단말노드 언어모델 기반의 XML 문서검색에서 구조 제한의 유용성에 관한 실험적 연구*

A Experimental Study on the Usefulness of Structure Hints in the Leaf Node Language Model-Based XML Document Retrieval

정 영 미(Youngmi Jung)**

초 록

XML 웹 문서 포맷은 문헌 내에 내용뿐만 아니라 의미 있는 논리적인 구조 정보를 포함할 수 있어, 검색에서 문서의 내용 뿐만 아니라 구조로 접근하는 것을 제공한다. 그래서 본 연구의 목적은 XML 검색에 있어 내용 검색에 추가적인 요소로 사용된 구조적인 제한이 얼마나 유용한지를 실험하기 위해 내용만으로 검색한 결과와 내용과 구조적인 제한을 가지고 검색한 결과간의 성능을 비교하였다. 이 실험은 자체 개발된 단말노드 언어모델기반의 XML 검색시스템을 사용하였고 INEX 2005의 ad-hoc트랙에 참여하여 모든 실험방법과 INEX 2005의 실험 문헌 집단을 사용하였다.

ABSTRACT

XML documents format on the Web provides a mechanism to impose their content and logical structure information. Therefore, an XML processor provides access to their content and strucure. The purpose of this study is to investigate the usefulness of structural hints in the leaf node language model-based XML document retrieval. In order to this purpose, this experiment tested the performances of the leaf node language model-based XML retrieval system to compare the queries for a topic containing only content-only constraints and both content constrains and structure constraints. A newly designed and implemented leaf node language model-based XML retrieval system was used. And we participated in the ad-hoc track of INEX 2005 and conducted an experiment using a large-scale XML test collection provided by INEX 2005.

키워드: XML 검색시스템, 언어모델, 구조 검색, 단말노드, 내용검색
XML retrieval system, language model, structured retrieval, leaf node, content retreival

---

* 본 연구는 2005년도 한국학술진흥재단의 신진연구인력연구장려금 지원에 의한 것임

** 경북대학교 문헌정보학과 강사(yomjung74@hanmail.net)

1. 서 론

차세대 웹 문서표준 XML(eXtensible Markup Language)은 웹의 광범위한 확산으로 인한 웹상의 다양한 분야의 이질적인 문서들 또는 이기종 시스템 구축되어 있는 다양한 문서들을 통합하고 자유롭게 교환하기 위해 1996년에 제안되었다. XML 표준은 이렇게 확장성과 호환성이 좋을 뿐만 아니라 의미있는 구조정보를 문서 내에 포함할 수 있기 때문에 기하급수적으로 증가한 웹 문서의 검색을 위해 좀 더 정확하고 특정적인 검색을 위한 프레임웍을 제공한다. 이러한 XML 문서의 장점은 다양한 분야의 다양한 속성을 지닌 XML 문서의 급속한 확장을 초래하였고, 이질적이고 수많은 구조화된 XML 문서를 효율적으로 저장하고 검색하기 위한 많은 연구들이 필요하게 되었다. 이들 상당수는 특히 전 세계적인 XML 검색 연구 그룹인 INEX(INitiative for the Evaluation of XML retrieval)를 통해 수행되고 있다.

XML 표준이 채택된 이후 수행된 XML 관련 연구들은 많은 부분이 XML 문서의 효율적인 저장구조 및 색인 기술에 관한 데이터베이스 지향적인 것(Bos, 1997; Turau, 1999; 박종관 외, 2001; 김은정, 2002)에 할당되어 왔다 최근에는 INEX를 중심으로 내용뿐만 아니라 구조 정보를 검색하기 위한 검색기법 및 시스템 개발, 그리고 그것의 성능평가에 관한 정보 검색 지향적인 연구(Ogilvie and Callan, 2003; 김희섭, 2004)와 더불어 구조 검색을 위한 XML 질의 언어에 관한 하이브리드 형 연구들(Wiegand, 2002; 정영미 외, 2005)도 점점 양적·질적으로 성장하고 있다.

하지만 XML 표준의 급속한 성장과 구조적인 정보를 이용한 효율적인 검색에 대한 상당한 기대감에도 불구하고 실 운용상에 있는 대규모의 XML 데이터베이스나 검색시스템의 성장 속도는 그 예상했던 것에 미치지 못한다. 물론 여기에는 비 구조화된 기존 문헌들을 구조화된 문헌으로 변환하는데 필요한 초기 비용의 부담감, XML의 구조정보를 효율적으로 활용할 수 있는 검색 기술의 미비, 그리고 구조 정보 검색에 대해 친숙하지 못한 이용자 요인, 이를 해소할 수 있는 효율적인 인터페이스 부재 등 다양한 요소들이 관련될 것이다.

또한 Sigurbjörnsson과 Kamps은 그들의 2004년의 연구에서 내용만으로 제한한 검색이 내용과 구조적인 정보를 모두 포함한 검색보다 평균 정확률이 더 나은 것으로 나타났다고 보고하고 있다(Sigurbjörnsson and Kamps, 2004). 물론 이런 결과에도 여러 요

인들이 관여했을 가능성이 있지만, 이것은 XML 문서의 검색에 있어서의 주요 장점인 내용뿐만 아니라 구조 검색을 통해 좀 더 정확하고 성능이 좋은 검색을 할 수 있다(Robert W. P. Luk et al, 2002)는 것과 대립된다. 이런 XML 문서에 관련된 근본적인 문제에 대해서는 정확하고 객관적인 더 많은 연구들을 통해 검토하는 것이 시급하다.

따라서 이 연구는 XML 문서의 성장에 관련된 상기의 다른 다양한 요인들은 배제하더라도, 현 상황에서 XML 문서에 포함된 구조적인 정보가 검색에서 얼마나 유용한지에 대한 실험함으로써 구조화된 XML 문서 검색의 쟁점에 대한 근본적인 문제에 접근하고자 하였다. 이를 위해 실험은 자체 개발한 단말노드(leaf node) 언어모델 기반의 XML 검색시스템을 사용하여 XML 문서에 대해 같은 토픽(topic)으로 구성된 내용검색(CO: content-only)과 구조적인 제한이 추가된 내용구조검색(CO+S: content only with structure hints)을 수행하여 이 둘 검색결과에 대한 성능비교를 하였다. 이 실험에서 사용된 실험문헌 집단은 INEX 2005에서 제공한 것이며, 실험의 모든 과정이 INEX 2005를 통해 수행되었기 때문에 이 실험에서 사용된 시스템의 구조 검색의 성능평가 뿐만 아니라 50여 개의 다른 참여 그룹들에서 실험한 다양한 검색기법 및 모델과의 상대적인 비교도 가능하다. 그래서 본 논문에서는 이 실험의 목적에 초점을 맞추기 위해 통제된 실험환경으로 구축된 XML 문서의 저장구조, 검색기법, 질의 처리, XML 질의 언어, 인터페이스 등에 대해서는 간략하게 기술할 것이다.

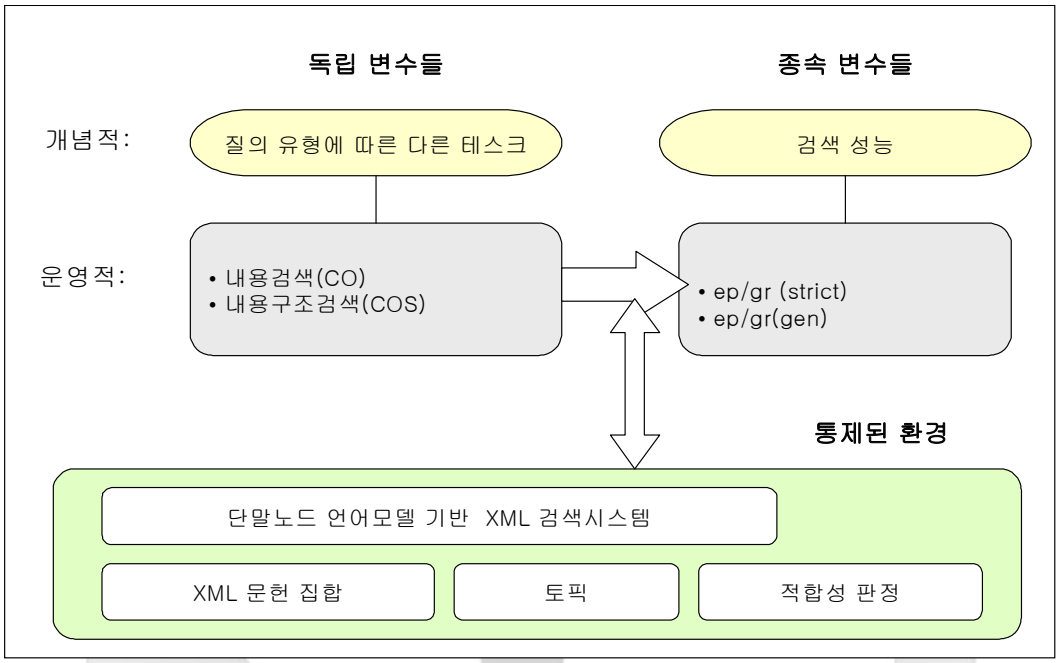
2. 실험 설계

2.1 실험 설계 개요

이 실험의 목적은 단말노드 언어모델 기반 XML 검색시스템을 사용하여 XML 문헌 검색에서 구조제한 검색이 내용 검색보다 얼마나 더 유용한지를 알고자 함에 있다. 이를 위한 실험의 전체적인 개요는 <그림 1>과 같다.

이 실험은 독립변수로 내용만으로 구성된 질의를 통한 내용검색과 내용과 구조적인 제한을 포함하고 있는 질의를 통한 내용구조검색의 두 가지 태스크를 설정하였고, 종속변수로 이들 태스크 각각에 대한 성능평가 측정방법을 설정하였는데 적합성 수준에 따라 부분 점수를 부여하지 않은 ep/gr(strict)와 ep/gr(gen) 측정의 두 가지의 측면 모두에서 살펴보고자 하였다. 이 모든 실험은 단말노드 언어모델 기반의 XML 검색시스템을 통해 수행되었고 사용한 실험문헌 집단은 INEX 2005를 통해 생성되어 제공된 것

이다.



<그림 1> 실험 설계 개요

2.2 통제된 환경: 검색시스템과 실험문헌 집단

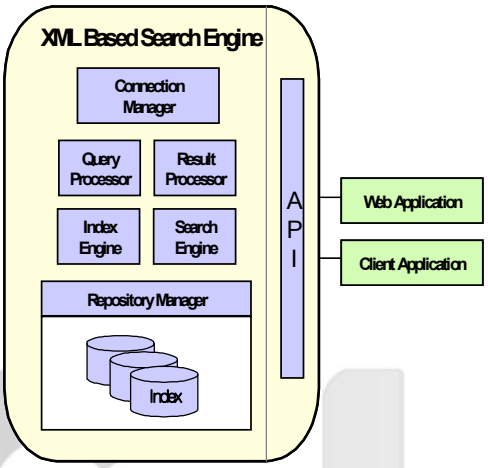
2.2.1 단말노드 언어모델 기반 XML 검색시스템

실험을 위해 두 가지 검색 태스크의 통제된 환경으로 단말노드 언어모델 기반 XML 검색시스템이 설계되고 구축되었다. 자연언어처리(NLP: Natural Language Processing) 분야에서 오랫동안 연구되어온 언어모델(LM: Language Model)은 Ponte와 Croft에 의해 1998년에 정보검색분야에 처음으로 소개된 이래, 몇몇 연구들에서 주목할 만한 우수한 성능을 가지는 것으로 나타났다(Ponte and Croft, 1998; Miller et al, 1999; Zhai and Lafferty, 2001). 초기의 이 분야 연구들의 대부분은 비구조화된 문헌 검색에 관한 것이었고 벡터모델이나 확률모델과 같은 전통적인 모델과의 성능 차이에 초점을 맞추고 있었으나, 최근에는 몇몇 연구들에서 내용뿐만 아니라 구조검색을 지원하기 위해 구조화된 XML 문헌 검색에 적용(Ogilvie and Callan, 2003; Sigurbjörnsson, Kamps, and Rijke, 2004)하거나 좀 더

나은 성능을 위해 추가적인 새로운 기법들을 도입(Zaragoza, Hiemstra, and Tipping, 2004; Zhai and Lafferty, 2004)하는 단계에까지 이르렀다(정영미, 김희섭, 2005). 이 실험에서 사용된 검색시스템도 내용뿐만 아니라 구조적인 검색까지 지원하는 XML 검색 시스템으로 비교적 단순한 언어모델링을 사용하였다.

① 시스템 개요

XML 문서를 저장 및 검색하기 위한 개발한 검색 시스템(Litch Search Server라 명함)의 전체 구조는 <그림 2>와 같이 API, 외부 시스템 또는 애플리케이션이 검색 엔진에 색인/검색 요청을 할 수 있도록 접속관리를 담당하는 Connection Manager, Connection Manager를 통해 전달된 질의를 분석하여 그에 따른 요청을 Index Engine 또는 Search Engine에 전달해주는 역할을 담당하는 Query Processor, 검색 대상이 되는 XML 문서를 처리하여 검색이 가능하도록 구조적으로 저장하는 Index Engine, Query Processor를 통하여 전달된 XQuery에 대한 내용을 단계별로 실행하는 Search Engine, Query Processor를 통하여 각각의 엔진 또는 Processor에 전달되어 생성된 결과 정보를 Connection Manager에 보낼 수 있도록 결과 정보에 대한 재구성하는 Result Processor, 그리고 Index Engine에서 처리된 XML 구조 정보를 Storage에 저장시키고 불러오는 작업을 담당하는 Repository Manager로 구성되어 있다.



<그림 2> 시스템 구조

② 색인작성

색인작성은 SAX 기반 Parser를 사용하여 이루어졌고 문서의 구조정보를 PreOrder 순서로 순회하면서 각각의 엘리먼트(Element)와 애트리뷰트(Attribute)의 값을 읽어서 각각의 테이블에 저장하였다. XPath는 Path Table에, 엘리먼트와 애트리뷰트의 이름 및 PreOrder에 해당하는 순서 정보는 Dom Index Table에, 그리고 엘리먼트와 애트리뷰트의 값은 스태밍과 불용어를 처리하여 저장하였다.

색인처리가 끝난 후 처리된 문서의 정보는 각각 색인기반 테이블과 B+Tree의 두 가지 형태로 저장되었다.

색인기반 테이블에서는 주로 문서의 ID를 관리하는 테이블이 저장되어 있으며 빠른 저장 및 검색을 필요로 하는 테이블은 B+ Tree에 저장하였다. 색인기반 테이블은 문서의 고유한 ID를 생성하여 저장하는 Document Table, XML 문서의 Path 정보를 ID별로 관리하는 PathIndex Table, 그리고 검색 단어를 ID로 저장하여 관리하는 Term Table로 구성되었고 Term에 대한 위치와 노드 정보들이 들어 있는 TermIndex Table과 문서의 구조 정보가 저장되어 있는 DOMTable은 B+ Tree에 저장되었다. <표 1>은 본 시스템의 실험을 위해 구축된 색인 테이블과 저장 유형을 보여준다.

<표 1> 색인 테이블과 저장 유형

Term Name	Storage Type
Document Table PathIndex Table Term Table	Index Table
TermIdex Table	B+ Tree
DOMTable	B+ Tree

③ 질의 처리

이 시스템에서는 구조적인 검색을 지원하기 위해 XML 질의 언어로 XQuery를 사용하였다. 검색 요청 구문이 입력되면 Search Engine은 <그림 3>와 같은 XQuery의 'For' 구문에 있는 $t의 값인 /books/*/article을 Basis로, XPathString을 키(Key) 값으로 DOM Table을 통해 XPathString에 속하는 Node List를 가지고 온다. 'Where'에 있는 복합 질의문을 단일 질의문으로 나눈 후 단일 질의문을 하나씩 실행하면서 연산을 수행한다. 단일 질의문에 있는 XPath 값을 이용하여 검색 Term Table에 해당하는 Node List를 얻어온 후 'for '에서 계산한 Node List와 비교하여 서로 일치되는 값들만 골라낸다. 이런 과정을 통해 'Where ' 구문만으로도 최종 검색 정보를 얻을 수 있다.

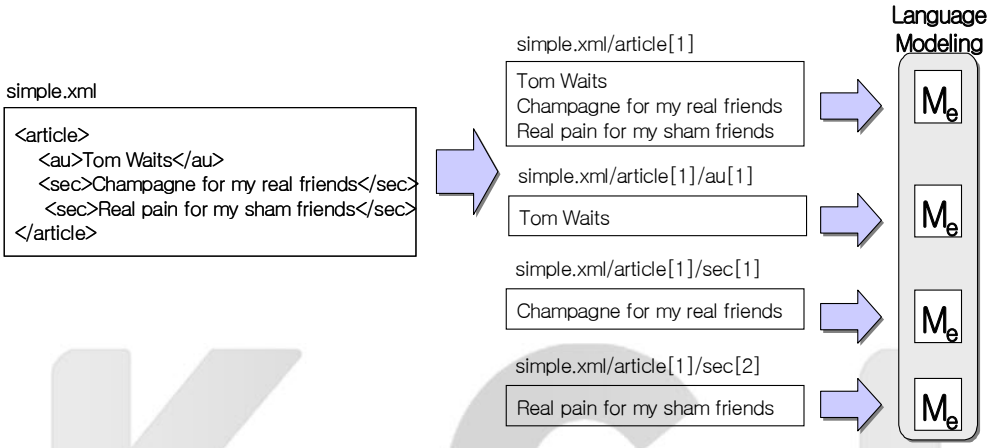
```
xquery

for $t in /article[/*/au/*/#TEXT = 'moldovan']
for $v in /article[/*/#TEXT = 'semantic networks construction use']
where $t and $v
order by DF() DESC
  <res>{$t*}</res>
```

<그림 3> XQuery의 예

④ 순위화 방법

앞에서도 언급했듯이, 결과로 제출된 리스트의 순위화를 위해 이 시스템에서 사용한 개념적 검색 모델은 Uni-gram 기반의 통계적 언어모델(Statistic Language Model)이다. XML 문헌과 같이 구조화된 문헌의 검색에서는 <그림 4>와 같이 그 검색단위가 엘리먼트(element)가 되기 때문에 언어모델링 측정 단위 또한 문헌이 아니라 문헌 속에 포함된 엘리먼트가

된다. 그래서 <그림 4>의 simple.xml /article[1]의 첫 줄과 simple.xml /article[1]/au[1]이 중복되어 측정되어지고 이것은 계층적으로 구조화된 대규모의 XML 문서를 문헌집합으로 하는 이 실험에서는 중복적인 아주 많은 작업과 저장 공간을 필요로 한다.



<그림 4> 구조화된 문헌의 언어모델링 단위

그래서 이 실험에서는 구조 제한 검색을 위해 각 엘리먼트의 언어모델을 다 측정하지 않고 단말노드를 단위로 언어모델을 측정하였다. 이것은 수식 (1)에서 보여진다.

$$P(E_{leaf}, Q) \propto P(Q|M_{E_{leaf}}) = \prod_{w \in (q_1, \ldots q_n)} P(w \mid M_{E_{leaf}})$$

(1)

E_{leaf}: 단어노드인 엘리먼트
Q : 질의
$M_{E_{leaf}}$: 단말노드의 언어모델

단말노드란 이 실험에서 사용되는 모든 문헌 집합은 계층적으로 구조화된 XML 문서로 트리구조로 표현 가능한데 원래는 이런 트리의 말단에 위치한 노드를 말한다. 하지만 여기에서는 검색의 최소단위가 될 수 있는 '/tit', '/abs', '/p', 그리고 '/ref' 등을 단말노드로 재구성 하였다. 왜냐하면, 특히 '/p'(paragraph) 아래에는 각종 애트리뷰트들과 ' /pp'와 같은 하위 paragraph으로 구성되어 있지만 그 속에 포함된 텍스트의 길이가 너무 짧아 적합한 정보로 판정하기 어렵기 때문이다. 그리고 단말노드가 아닌 루트 노드를 포함한 부모 노드들이 검색 단위가 되는 경우에는 그 부

모 노드 속에 포함된 단말노드들의 확률 값과 단말노드의 언어모델 값을 결합하여 부모 노드의 언어모델 값을 추정하였다. 이렇게 단말노드를 기반으로 측정되고 추정된 엘리먼트들의 언어모델은 질의를 생성할 확률 $P(E, Q)$에 의해 순위화 된다. 그것은 수식(2)에서 보여진다.

$$P(E_{parent}, Q) \propto P(E_{child1}) \cdot P(Q|M_{E_{child1}}) + P(E_{child2}) \cdot P(Q|M_{E_{child2}}), ... + P(E_{child_n}) \cdot P(Q|M_{E_{child_n}}) \quad (2)$$

E_{parent}: 부모 노드(엘리먼트)
E_{child_n}: n번째 자식노드
$M_{E_{child_n}}$: n번째 자식노드의 언어모델

단, 이 시스템에서 부모 노드에 대한 자식 노드들의 확률 값을 측정하기 위해 각 노드 속에 포함된 텍스트의 길이 대신 저장 용량을 사용하였다. 그래서 엘리먼트의 확률 $P(E_{child})$는 수식(3)에 의해 보여진다.

$$P(E_{child}) = \frac{|E_{child}|}{|E_{parent}|} \quad (3)$$

2.2.2 실험 문헌 집단

이 실험에서 사용된 실험 문헌 집단은 INEX 2005의 참여를 통해 생성되고 제공된 것이다. INEX는 전 세계적인 XML 검색을 위한 연구 그룹들의 모임으로 대규모의 문헌 집합과 토픽 집합, 각 토픽에 대한 적합한 문헌 집합을 생성하여 제공한다. 이용자의 요구를 대신하는 인공적인 토픽과 각 토픽에 대한 적합한 문헌 집합은 참여자들에 의해 제출되고 판단되어 생성된다. INEX 2005에는 Cornell University, Rutgers University, City University London, University of California, Berkely, IBM Haifa Research Lab 등 전 세계의 대학 및 연구소의 60개 연구팀이 참여하였다.

① 문헌 집합

이 실험에서 사용된 INEX 2005의 문헌집합은 IEEE Computer Society의 1995년부터 2004년까지의 저널에 수록된 16,819개의 기사논문으로 구성되어 있다. 문헌집합내의 문서는 총 192개의 다른 DTD로 구성되어 있고 다양한 길이로 이루어져 있다. 하나의 문서는 평균 1,532개의 XML 노드를 가지고 노드의 평균 깊이는 6.9이다. 그래서 이 문헌 집합으로 검색 가능한 엘리먼트는 10만개 이상이고 이 문헌집합의 총 용량은 764MB이다.

② 토픽 집합

INEX 2005에서는 40개의 CO+S(Content Only with Structure Hints) 토픽과 46개의 CAS(Content And Structure) 토픽을 제공했다. 이 실험을 위해서는 <그림 5>와 같은 CO+S 토픽이 사용되었다. 토픽의 구

성은 질의어가 되는 <title>과 <castitle>이 있고 자연어 처리에 대

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<!DOCTYPE inex_topic SYSTEM "topic.dtd" >
<inex_topic topic_id="202"  query_type="CO+S" ct_no="1" >
<InitialTopicStatement>I'm interested in knowing how ontologies are used to encode knowledge in real world scenarios. I'm writing a report on the use of ontologies. I'm particularly interested in knowing what sort or concepts and relations people use in their ontologies. </InitialTopicStatement>
<title>ontologies case study</title>
<castitle>//article[about(., ontologies)]//sec[about(., ontologies case study)]</castitle>
<description>Case studies in the use of ontologies</description>
<narrative>I'm writing a report on the use of ontologies. I'm interested in knowing how ontologies are used to encode knowledge in real world scenarios.  I'm particularly interested in knowing what sort or concepts and relations people use in their ontologies. I'm not interested in general ontology frameworks or technical details about tools for ontology creation or management. An example relevant result contains a description of the real world phenomena described by the ontology and also lists some of the concepts used and relations between concepts.
</narrative>
</inex_topic>
```

<그림 5> CO+S 토픽의 예: INEX 2005의 topic-id='202'

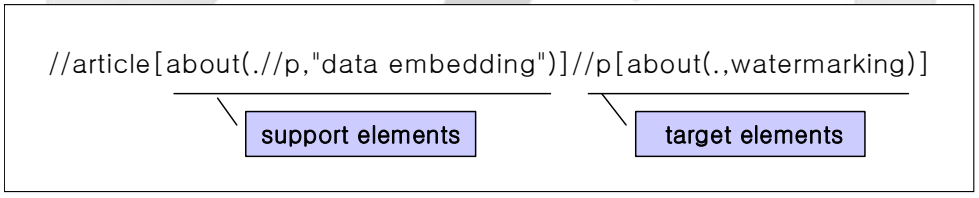
한 실험을 제공하기 위해 좀더 상세하게 기술된 <description>, 그리고 적합성 판정을 위해 활용할 수 있도록 질의를 좀 더 명확하게 정의해 놓고 있는 <narrative>로 이루어져 있다. CO+S 토픽의 <title>은 내용만을 위한 질의를 나타내고 <castitle>은 이런 내용에 추가적으로 구조적인 제한을 지니는 질의를 나타낸다. <castitle>의 구조 제한에 대한 표현은 NEXI (Narrowed Extended XPath I)를 따르고 있으며 이것은 이 실험에서 XQuery로 변환되어 사용된다.

두 태스크를 위한 <title>과 <castitle>의 몇 가지를 예를 보면 <표 2>와 같다.

CO+S 토픽 중 몇몇 <castitle>은 두 가지 이상의 중첩적인 구조 제한을 표현하고 있는데, 그 예는 <그림 6>와 같다. 예에서 //article[about(.//p, "data embedding")]는 지원 엘리먼트(support element)로 이것은 이용자에게 검색결과로 보여주기 위한 엘리먼트가 아니라 일단 “data embedding" 이라는 검색어에 대한 'paragraph'이 포함된 'article'로 제한하여, 뒤에 따라 나오는 표적 엘리먼트(target element)인 //article//p[about., watermarking)]에 대한 검색 결과 값으로 돌려달라는 것이다. 이 실험에서 사용된 시스템은 다른 경로를 지니는 중첩질의에 대한 처리를 수행하지 못함으로 표적 엘리먼트에 한해 검색을 수행하였다.

<표 2> CO+S 토픽의 <title>과 <castitle> 예

Topic ID	Tag	CO+S Topic
202	<title>	ontologies case study
	<castitle>	//article[about(., ontologies)]//sec[about(., ontologies case study)]
203	<title>	code signing verification
	<castitle>	//sec[about(., code signing verification)]
204	<title>	moldovan semantic networks
	<castitle>	//*[about(.//au, moldovan) and about(., "semantic networks")]
216	<title>	multimedia retrieval system architecture
	<castitle>	//sec[about(., multimedia retrieval system architecture) or about(.//fig, multimedia retrieval architecture)]
232	<title>	Dempster Shafer theory Database experiment
	<castitle>	//article[about(.//abs, Dempster-Shafer theory)]//sec[about(., Dempster Shafer database experiment)]



<그림 6> 지원 엘리먼트와 표적 엘리먼트로 구성된 질의 예

③ 토픽에 대한 적합문헌 집합

참여자들에 의해 제출된 결과들은 토픽별로 풀링(pooling)되어 저장되고 각 토픽별 결과세트들은 각 참여자들에게 2~3개씩 할당되어 INEX의 온라인 평가 시스템인 X-Rai를 통해 적합성이 평가되었다. 토픽에 대한 결과 문헌의 적합성 평가는 토픽에 대해 문헌의 구성요소들 전반에 걸쳐 얼마나 광범위하게 논의하고 있는지에 대한 망라성(e: Exhaustivity)과 토픽에 얼마나 초점을 맞추고 있는지에 대한 특정성(s: Specificity)의 두 가지 측면에서 평가되었다. 이 두 가지 측면은 각각 $e \in [?,0,1,2]$, $s \in [o,1]$로 평가되고 <그림 7>은 적합성이 판정된 topic id='234'에 대한 적합한 문헌 집합의 일부를 보여준다.

```
<file collection="ieee" name="mu/1995/u2030">
  <passage start="/article[1]/bdy[1]/sec[3]/ss1[1]/p[1]/text()[1].0" end="/article[1]/bdy[1]/sec[3]/ss1
[1]/p[6]/text()[1].470" size="3146"/>
  <element path="/article[1]/bdy[1]/sec[3]/ss1[1]" exhaustivity="2" size="3181" rsize="3146"/>
  <element path="/article[1]/bdy[1]/sec[3]" exhaustivity="2" size="25636" rsize="3146"/>
  <element path="/article[1]/bdy[1]" exhaustivity="2" size="36512" rsize="3146"/>
  <element path="/article[1]" exhaustivity="2" size="41586" rsize="3146"/>
  <element path="/article[1]/bdy[1]/sec[3]/ss1[1]/p[1]" exhaustivity="2" size="1001" rsize="1001"/>
  <element path="/article[1]/bdy[1]/sec[3]/ss1[1]/p[2]" exhaustivity="1" size="287" rsize="287"/>
  <element path="/article[1]/bdy[1]/sec[3]/ss1[1]/p[3]" exhaustivity="1" size="723" rsize="723"/>
  <element path="/article[1]/bdy[1]/sec[3]/ss1[1]/p[4]" exhaustivity="1" size="310" rsize="310"/>
  <element path="/article[1]/bdy[1]/sec[3]/ss1[1]/p[5]" exhaustivity="1" size="359" rsize="359"/>
  <element path="/article[1]/bdy[1]/sec[3]/ss1[1]/p[6]" exhaustivity="1" size="471" rsize="471"/>
```

<그림 7> 적합성 판정이 된 결과 리스트의 예

2.3 독립 변수: 태스크들

구조화된 XML 문헌의 검색에서 내용 검색에 추가적으로 부여된 구조적인 제한이 얼마나 유용한지를 실험하기 위해 두 가지의 태스크가 수행되었다. <표 3>에서와 같이, 그 중 하나는 CO 태스크로, CO+S 토픽의 <title> 태그 내의 내용만으로 구성된 질의어를 사용하여 검색하였고 나머지 하나는 COS 태스크로 CO+S 토픽의 <castitle> 내의 내용구조질의어를 사용하여 검색한 것이다.

<표 3> 독립변수: 태스크들

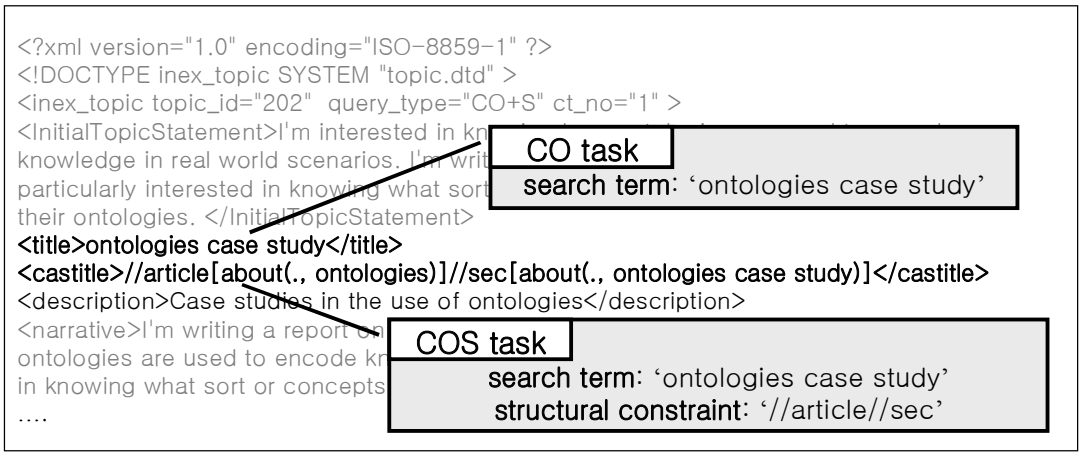
Task	Topic	Topic Tag	Run ID
CO	CO+S	<title>	CO.Thorough
COS	CO+S	<castitle>	COS.Thorough

검색전략은 두 태스크 모두 해당되는 모든 엘리먼트를 검색하는 Thorough를 사용하였다.

<그림 8>은 INEX 2005의 CO+S topic-id='202'에서 CO와 COS 태스크를 위해 질의를 생성한 예를 보여준다.

2.4 종속 변수: 검색 성능 측정

INEX 2005에서는 각 참여팀들의 제출 검색결과에 대해 통일된 검색 성능 평가 지표를 통해 객관적으로 성능을 측정해주고 그 값을 통해 참여팀들의 상대적인 순위를 결정해준다. INEX 2005에서 사용된 평가 지표로 <표 4>와 같이 ep/gr (effort-precision /gain-recall)이 사용되었다.



<그림 8> CO 태스크와 COS 태스크의 각각의 질의 예: topic-id='202'

<표 4> 검색 성능 측정 방법

Performance Measure	Metric	Quantization Function
System-oriented measure	ep/gr (overlap=off, strict)	$quant_{strict}$
	ep/gr (overlap=off, gen)	$quant_{gen}$

이 두 가지의 측정방법 모두 망라성과 특정성의 적합성 판정에 따라 그 값을 어떻게 부여하는가에 따라 'strict'와 'gen (generalized)'한 것으로 다시 세분되었다. 이것은 수식 (4)와 (5)를 통해 각각 계산되었다.

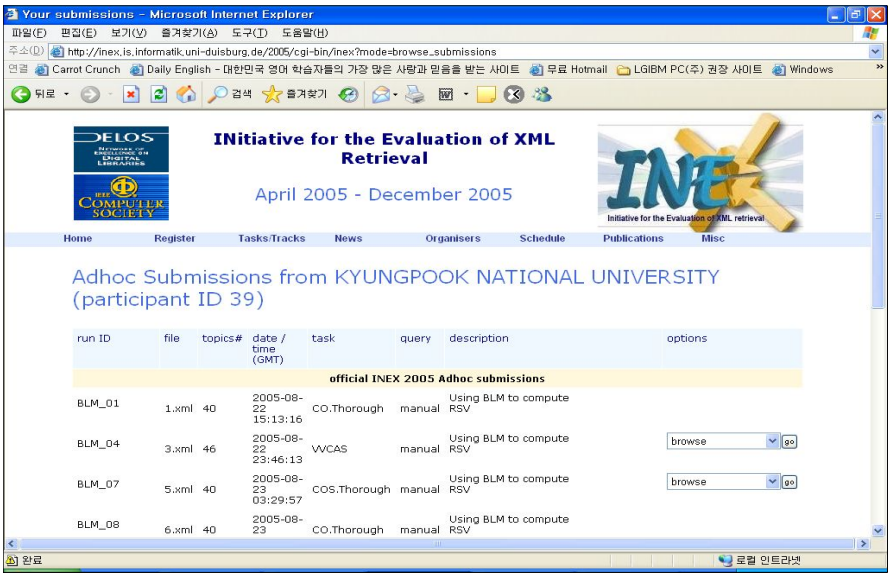
$$strict(e,s):\begin{cases}1 & \text{if } e=2 \text{ and } s=1,\\ 0 & otherwise.\end{cases} \quad (4)$$

$$gen(e,s) := e \times s \quad (5)$$

3. 실험 및 결과분석

3.1 실험

이 연구의 목적을 위해 CO 태스크와 COS 태스크가 수행되었고 그 결과가 <그림 9>와 같이 INEX 2005에 제출되었다. 제출된 검색결과에는 <그림 10>과 같이, 파일명과 엘러먼트의 경로, 그리고 RSV (Retrieval Status Values)에 의한 순위 정보가 포함되어 있다.



<그림 9> INEX 2005의 온라인 Submission Tool

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE inex-submission SYSTEM "http://inex.is.informatik.uni-duisburg.de/2005/
submission.dtd">
<inex-submission participant-id="39" run-id="BLM_title_01" task="CO.Thorough"
query="manual">
    <description>Using BLM to compute RSV</description>
    <topic topic-id="202">
        <collections>
            <collection>ieee</collection>
        </collections>
        <result>
            <file>ts/2002/e0721</file>
            <path>/article[1]/bdy[1]/sec[2]/ss1[2]/ss2[1]/list[2]/item[2]/p[1]</path>
            <rank>1</rank>
        </result>
        <result>
            <file>tk/1999/k0464</file>
            <path>/article[1]/bdy[1]/sec[1]/list[1]/item[1]/p[1]</path>
            <rank>2</rank>
        </result>
        [ . . . ]
    </topic>
    <topic topic-id="203">
        [ . . . ]
    </topic>
</inex-submission>
```

<그림 10> INEX 2005에 제출된 결과세트 예

CO 태스크와 COS 태스크 모두 한 토픽 당 1500건 이하의 결과만을 제출할 수 있었고, CO 태스크는 각 토픽에 대해 검색한 결과, 평균 결과 건수가 527로 나타났고, COS 태스크는 평균 160개의 결과가 검색되었다. COS 태스크는 CO 태스크의 내용만을 검색한 결과에 대해 구조적인 제한으로 검색 결과를 제한했기 때문에 당연한 결과이다.

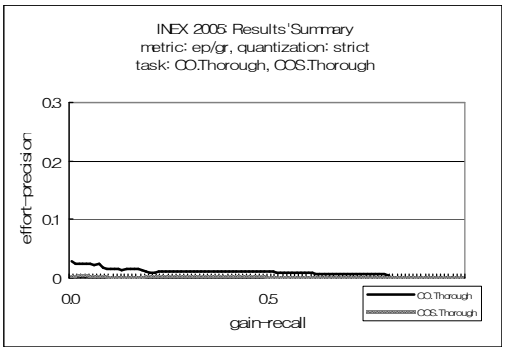
3.2 실험 결과 분석

3.2.1 단말노드 언어모델에서 구조 제한 검색의 유용성

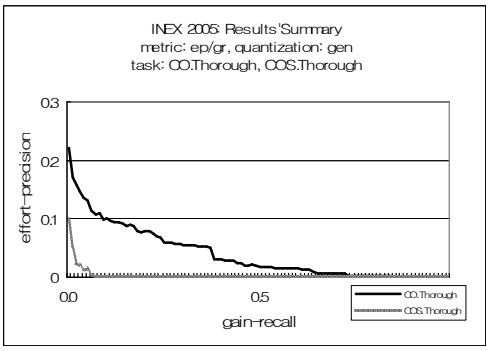
단말노드 언어모델 기반의 XML 검색 시스템을 통해 검색되어 INEX 제출된 CO와 COS 태스크는 시스템 지향적인 측정 방법 ep/gr의 'strict'와 'gen'의 두 가지 측면에서 해석되어 계산되었다. 중복적인 결과에 대해서는 계산되지 않았다. 그 결과가 <표 5>와 같이, ep/gr(strict)와 ep/gr(gen)의 두 가지 측면에서 모두 내용만을 검색한 CO 태스크가 모두 높게 나타났다. 망라성의 수준이 가장 높은 경우에만 점수를 부여한 ep/gr(strict) 측정에서는 평균 'effort-precision ' 값인 MAep가 CO 태스크 0.0126, COS 태스크 0.0014로 CO 태스크에 비해 COS 태스크의 성능이 10%의 결과에도 미치지 못하는 반면에, 망라성의 수준에 따라 점수를 부여한 ep/gr(gen)은 CO 태스크는 MAep = 0.0340, COS 태스크는 MAep = 0.0091로 그 차이가 줄어들었음을 볼 수 있다. 이것은 대부분의 COS 태스크의 결과가 구조적인 제한을 가짐으로 인해 문헌의 길이가 CO 태스크의

<표 5> 단말노드 언어모델의 CO.Thorough와 COS.Thorough의 성능: ep/gr

Run ID	ep/gr (Overlap=off, strict)				
	gr=0.01	gr=0.02	gr=0.1	gr=0.2	MAep
CO.Thorough	0.0293	0.0250	0.0176	0.0126	0.0126
COS.Thorough	0.0026	0.0023	0.0013	0.0010	0.0014
	ep/gr (Overlap=off, gen)				
	gr=0.01	gr=0.02	gr=0.1	gr=0.2	MAep
CO.Thorough	0.2208	0.1718	0.1101	0.0799	0.0340
COS.Thorough	0.0979	0.0509	0.0009	0.0002	0.0091



<그림 11> 성능비교: ep/gr(strict)



<그림 12> 성능비교: ep/gr(gen)

결과보다 짧게 나타났기 때문에 문헌에서 주제에 대한 망라적 논의에 대한 값이 부분점수를 가진 경우가 많기 때문이다. <그림 11>과 <그림 12>는 각각 ep/gr(strict)과 ep/gr(gen)에 대한 그래프를 비교해서 보여준다.

즉 이 실험에서 설계되고 구축된 단말노드 언어모델 기반의 XML 검색 시스템상에서는 내용만으로 구성된 질의로 검색을 수행하는 것이 구조적인 제한을 두는 것보다 더 성능이 우수한 것으로 나타났기 때문에 단말노드 언어모델 상에서는 XML의 구조적인 제한 검색의 유용성은 없는 것으로 보여진다. 하지만 단말노드 언어모델이 검색 성능에 큰 영향요인 임으로 이런 결과로 XML 문서의 구조적인 유용성에 대한 결론을 내릴 수는 없다. 왜냐하면 비록 언어모델이 비구조화된 문헌의 검색이나 구조화된 문헌의 내용 검색에서 주목할 만한 성능을 보였지만 구조 제한 검색에서의 성능에 대해서는 그 연구가 미비하기 때문이다. 그래서 이런 결과의 원인이 구조 검색에 대한 단말노드 언어모델 자체의 문제인지 XML의 구조 검색의 문제인지를 살펴보기 위해 다음 섹션에서는 INEX 2005에 각기 다른 검색기법이나 모델로 구축된 시스템으로 참여하여 실험을 수행한 다른 연구팀들과의 결과를 상대적으로 비교해 보았다.

3.2.2 단말노드 언어모델 기반의 XML 검색 시스템의 상대적인 성능 평가

INEX 2005에 참여한 전 세계의 60개 연구팀은 CO 태스크에는 총 57개의 결과 세트를 제출하였고 COS 태스크에는 33개의 결과 세트만을 제출하였다. 단말노드 언어모델 기반의 XML 검색시스템을 사용하여 수행한, 이 실험을 통해 제출된 결과들의 성능 평가 값에 따라 INEX 2005 참여팀 내에서의 상대적인 위치 및 순위는 살펴보면 <표 6>

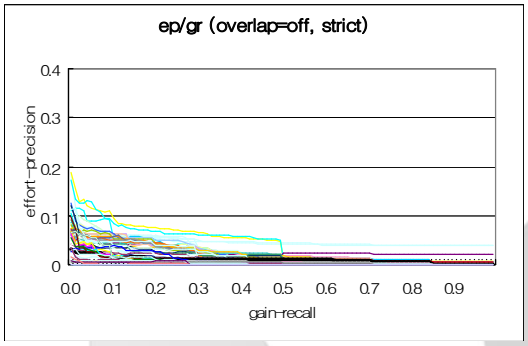
<표 6> INEX 2005 참여팀 내에서의 상대적인 위치 및 순위

Run ID	ep/gr (Overlap=off, strict)		
	total runs	our position	%
CO.Thorough	57	32	0.5614
COS.Thorough	33	29	0.8787
Run ID	ep/gr (Overlap=off, gen)		
	total runs	our position	%
CO.Thorough	57	37	0.6491
COS.Thorough	33	30	0.9090

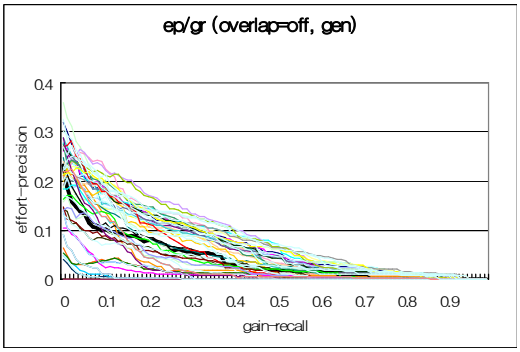
과 같다. 제출된 CO 태스크는 ep/gr(strict)에서 총 57개결과 중에서 32위를 차지하여 약 56%의 위치에 나타났고, COS 태스크는 33개결과 중에서 29위로 그 상대적인 위치가 약 87%로 나타났다. 망라성의 수준에 따라 부분적인 점수가 부여된 ep/gr(gen) 측정에서는 CO 태스크가 57개중 37위(약 65%)로 나타났고 COS 태스크는 33개중 30위(약 90%)로 성능이 좋지 못한 것으로 나타났다. 비교적 단순하고 중첩 질의를 수행하지 못했음에도 불구하고 단말노드 언어모델을 적용한 XML 시스템에서 내용만으로 검색한 CO 태스크는 상당히 주목할 만한 결과가 나타났다. 하지만 단말노드 언어모델은 COS 태스크에서 성능이 좋지 못한 것으로 나타났으며 이것은 단말노드 언어모델이 XML의 구조제한 검색을 잘 수행하지 못한다는 결론을 제시할 수 있다. 이런 요인이 CO 태스크와 COS 태스크간의 성능 차이를 더 현격하게 하는데 영향을 끼쳤다고 보여진다.

<그림 13>과 <그림 14>는 단말노드 언어모델 기반 시스템을 사용하여 XML 문헌 검색에 대해 내용만으로 접근한 CO 태스크에 대한 INEX 2005 참여팀 들과의 성능 비교 그래프이다. 그리고 <그림 15>와 <그림 16>은 이 실험에서 제출된 내용검색에 추가적으로 구조적인 제한에 둔 COS 태스크에 대한 INEX 2005 참여팀 들과의 성능 비교 그래프이다. 굵게 나타난 부분이 단말노드 언어모델 기반의 시스템을 사용하여 제출한 결과에 대한 성능 그래프이다. <그림 13>과 <그림 15>, 그리고 <그림 14>와 <그림 16>을 각각 비교해보면 알 수 있듯이 언어모델을 채택한 이 실험 외에도 INEX 2005에

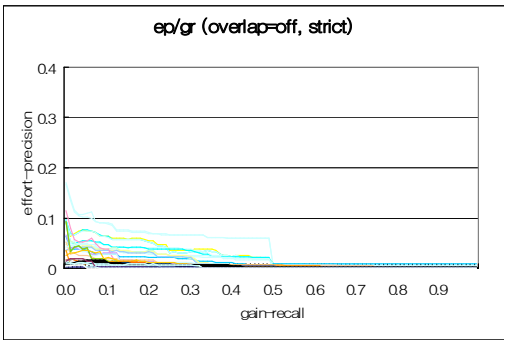
참여한 대부분의 연구팀 결과가 구조적인 제한을 둔 검색보다 내용검색이 더 우수한 성능을 나타냄을 알 수 있다. INEX 2005의 실험문헌 집단을 사용한 실험에 한해, 즉 어떤 검색모델을 사용하더라도 XML 검색에서 구조 제한 검색의 유용성이 없다는 결론을 내릴 수 있다.



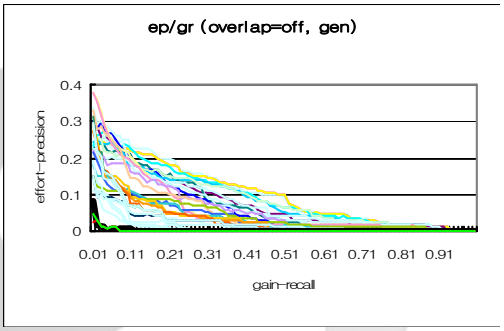
<그림 13> INEX 2005에서 CO.Thorough의 상대적인 성능 비교: ep/gr(strict)



<그림 14> INEX 2005에서 CO.Thorough의 상대적인 성능 비교: ep/gr(gen)



<그림 15> INEX 2005에서 COS.Thorough의 상대적인 성능 비교: ep/gr(strict)



<그림 16> INEX 2005에서 COS.Thorough의 상대적인 성능 비교: ep/gr(gen)

4. 결 론

XML 문서는 단순한 마크업 언어를 떠나 의미 있는 구조 정보를 포함하고 있어 문서내에 포함된 구조 정보를 활용하여 좀 더 정확하고 특정적인 검색을 제공하는 것으로 알려져 있다. 또한 언어모델은 1998년 Ponte와 Croft

에 의해 정보검색 분야에 소개된 이래 많은 연구들에서 주목할 만한 성능을 보이는 검색모델로 각광받고 있다. 물론 대부분의 연구가 비구조화된 문헌의 검색에 관한 것이고 그 중 몇몇이 구조화된 문헌에 대한 것이지만 내용검색에 관한 것이다.

이에 본 연구의 목적은 단말노드 언어모델 기반 XML 검색시스템 상에서 내용검색과 내용구조제한 검색의 성능을 비교해봄으로써 XML 문서의 구조정보가 이 시스템상의 검색에서 얼마나 유용한지를 살펴보기 위한 것이다. 이 연구는 객관적이고 상대적인 성능 평가를 위해 전 세계적인 규모의 INEX 2005 ad-hoc Track에 공식적으로 참여하여 여기에서 제공하는 대규모의 테스트 컬렉션을 사용하여 실험하였다. INEX 2005의 문헌 집합은 IEEE Computer Society에서 출판된 총 16,819개의 계층적으로 구조화된 XML 논문기사로 구성되어 있고 이 연구에서는 이것을 구조 검색이 가능하도록 색인기반 테이블과 B+ Tree의 저장유형으로 색인 테이블을 작성하였다. 또한 INEX 2005의 토픽 중 40개의 CO+S 토픽을 사용하여 <title>에 포함되어 있는 내용질의를 통해 CO 태스크를 수행하였고, <castitle>의 내용구조질의를 사용하여 COS 태스크를 수행하였다. 실험을 통해 발생된 이 두 가지 결과세트는 INEX 2005에 온라인으로 제출하였고, 공식적고 통일된 평가지표 ep/gr(strict)와 ep/gr(gen)에 의해 객관적으로 성능 평가 되었다.

단말노드 언어모델기반 XML 검색시스템에서는 모든 평가 지표에 대해 내용만으로 검색한 CO 태스크가 구조적인 제한을 가지는 COS 태스크보다 성능이 우수한 것으로 나타났고 그나마 망라성의 수준에 따라 부분적인 점수를 부여한 ep/gr(gen)에서 그 격차가 덜 한 것으로 나타났다. 이런 결과는 일단 단말노드 언어모델기반 XML 검색시스템에서는 부가적인 구조 제한검색의 유용성이 없는 것으로 볼 수 있다.

또한 단말노드 언어모델기반 XML 검색시스템을 사용한 이 두 가지 태스크에 대한 상대적인 성능 평가를 위해 다른 검색 기법이나 모델을 사용하여 결과를 제출한 INEX 2005의 참여팀과 성능을 비교하였다. 그 결과 단말노드 언어모델기반 XML 검색시스템을 사용한 CO 태스크의 성능은 ep/gr(strict) 평가 지표에 대해 32/57로 약 56%, ep/gr(gen)에서는 37/57로 약 65%로 주목해 볼만한 결과로 나타났다.. 반면 COS 태스크는 ep/gr(strict)와 ep/gr(gen)의 평가지표에 대해 각각 약 87%, 약 90%로 성능이 좋지 못한 것으로 나타났다. 이런 결과는 단말노드 언어모델이 내용검색과 달리 계층적으로 구조화된 XML 문서의 구조 제한 검색을 잘 수행하지

못하는 것으로 볼 수 있다.

게다가 INEX 2005에 참여한 다른 팀들의 구조 제한 검색의 유용성에 대한 결과를 살펴보면, CO 태스크에 제출된 결과는 총 57개임에 반해 COS에는 33개 결과만이 제출된 것으로 많은 팀들이 내용검색 만을 수행한 것을 알 수 있고 또한 CO와 COS 태스크를 모두 수행한 참여팀의 경우에도 COS가 CO 태스크 보다 성능이 우수하게 나타난 곳은 오직 한 곳밖에 없다. 즉 적어도 INEX 2005의 실험 문헌 집단을 사용한 실험에서는 단말노드 언어모델기반 XML 검색시스템을 사용하여 나타난 두 태스크의 현격한 성능 차이만큼은 아니지만 INEX 2005에 참여한 모든 연구팀들의 결과도 유사하게 나타남으로써 현재 수준의 XML 문헌의 구조 제한 검색은 그 유용성이 없다고 보여진다.

앞으로 XML 문서의 계층적인 구조를 실제의 검색에서 유용하게 사용하기 위해서는 이것에 관련된 문헌 자체의 요인, 검색시스템 요인, 이용자 관련 요인 등에 대한 후속적인 연구들이 절실히 필요하다.

참 고 문 헌

김희섭. 2004. Retrieval Performance of XML document Using Object-Relational Database. 『정보관리학회지』, 22(2): 189-210.

박종관. 2001. XML 문서의 효율적인 구조검색을 위한 색인모델. 『한국정보처리학회논문지』, 8(D): 451-460.

정영미, 김희섭. 2005. 정보검색에서의 언어모델 적용에 관한 분석. 『한국도서관·정보학회지』, 36(2): 49-68.

정영미 외. 2005. XQuery 기반 XML 검색시스템의 구조적인 질의 검색 성능평가. 『제12회 한국정보관리학회 학술대회 논문집』, 295-304.

Bos, B. 1997. “XML representation of a relational database.” [cited 2006. 11. 23] <http://www.w3.org/XML/RDB.html>

INitiative for the Evaluation of XML Retrieval 2004 homepage. [cited 2006. 10. 28] <http://inex.is.informatik.uni-duisburg.de:2004/index.html>

Miller, D., T. Leek, and R. Schwartz.

1999. “A Hidden Markov Model Information Retrieval System.” In *Proceedings of the 22nd Annual International ACM SIGIR Conference*, 214-221.

Ogilvie, Paul and J. Callan. 2003. “Language Models and Structured Document Retrieval.” In *Proceedings of the First Workshop of the INitiative for the Evaluation of XML Retrieval*, 12-18.

Ponte, Jay M. and W. Bruce Croft. 1998. “A Language Modeling Approach to Information Retrieval.” In *Proceedings of the 21nd Annual International ACM SIGIR Conference*, 275-281.

Robert, W. P. Luk et al. 2002. “A Survey in Indexing and Searching XML Document.” *Journal of The American Society for Information Science and Technology*, 53(6):415-437.

Sigurbjörnsson, Börkur, Jaap Kamps, and Maarten de Rijke. 2004. “An Element-Based Approach to Retrieval.” In *Proceedings of the 2003 Workshop of the INitiative for the Evaluation of XML Retrieval*, 19-26.

Turau, V. 1999. “Making legacy data accessible for XML application.” [cited 2006. 9. 1] <http://www.informatik,fh-wiesbaden.de/~turau/ps/legacy.pdf>

Wiegand, Nancy. 2002. “Investigating XQuery for Querying Across Database Object Types.” *SIGMOD Record*, 31(2): 28-33.

Zaragiza, H., D. Hiemstra, and M. Tipping. 2003. “Bayesian Extension to the Language Model for Ad Hoc Information Retrieval.” In *Proceedings of the 26nd Annual International ACM SIGIR Conference*, 4-9.

Zhai, C. and J. Lafferty. 2001. “Document Language Models, Query Models, and Risk Minimization for Information Retrieval.” In *Proceedings of the 24nd Annual International ACM SIGIR Conference*, 111-119.

Zhai, Chengxiang and John Lafferty. 2004. “A Study of Smoothing

Methods for Language Models Applied to Information Retrieval." *ACM Transactions on Information Systems,* 22(2): 179-214.