질의응답시스템 응답순위 개선을 위한 새로운 유사도 계산방법

(A New Similarity Measure for Improving Ranking in QA Systems)

김 명 관 † 박 영 택 ††
(Myung-Gwan Kim) (Young-Tack Park)

요 약 본 논문에서는 질의응답시스템의 성능을 개선하기 위해 문장의 위치정보와 질의형태분류기를 사용하여 질의에 대한 대답순위를 조정하는 새로운 질의-문서 유사도 계산을 제안한다. 이를 위해 첫째로 문서내용을 표현하고 문서의 위치정보를 반영하기 위해 개념그래프를 사용한다. 이 방법은 문서비교에 대표적으로 사용되는 Dice-Coefficient에 기반하고 문장에서 단어의 위치정보를 반영한 유사도 계산이다. 두 번째로 질의응답시스템의 대답순위를 개선하기 위하여 질의형태를 고려한 기계학습을 통한 질문에 대한 분류를 하였으며 이를 위해서 뉴스그룹의 FAQ 문서 30,000개를 가지고 기계학습 방법인 나이브 베이지안을 사용한 분류기를 구현하였다. 이에 대한 평가를 위해 세계적인 정보검색대회인 TREC-9의 질의응답시스템분야에 제출된 데이타를 가지고 실험하였으며 기존의 방법에 비해 자동학습기법을 사용하였음에도 평균상호순위가 0.29, 상위 5위에 정답을 포함시킨 경우가 55.1%의 성능을 보였다. 이 방법은 다른 시스템과 달리 질의형태분류를 기계학습 방법을 사용하여 자동으로 학습하는 것에 의의를 갖는다.

키워드 : 질의응답시스템, 기계학습

Abstract The main idea of this paper is to combine position information in sentence and query type classification to make the documents ranking to query more accessible. First, the use of conceptual graphs for the representation of document contents in information retrieval is discussed. The method is based on well-known strategies of text comparison, such as Dice Coefficient, with position-based weighted term. Second, we introduce a method for learning query type classification that improves the ability to retrieve answers to questions from Question Answering system. Proposed methods employ naive bayes classification in machine learning fields. And, we used a collection of approximately 30,000 question-answer pairs for training, obtained from Frequently Asked Question(FAQ) files on various subjects. The evaluation on a set of queries from international TREC-9 question answering track shows that the method with machine learning outperforms the underline other systems in TREC-9 (0.29 for mean reciprocal rank and 55.1% for precision).

Key words : Question Answer system, Machine Learning

1. 서 론

현재 개발되어 있는 대부분의 정보검색시스템은 문서단위의 검색을 지원하고 있다. 문서검색시스템은 사용자의 질의에 대해 관련 있는 문서들을 결과로 제시한다. 사용자의 질의가 구체적인 대답을 요구하는 것일 경우에는 문서검색시스템에서의 결과는 사용자가 문서를 읽고 원하는 대답을 찾아야 하는 불편함이 있다. 따라서 사용자의 질의에 대해 문서단위가 아니라 문서에서의 구체적인 대답을 검색할 수 있는 시스템에 대한 요구가 증가하고 있다. 사용자의 질의에 대해 구체적인 대답을 찾아주는 시스템을 질의응답시스템이라고 한다[1]. 국제적인 정보검색평가대회인 TREC(Text REtrieval Conference)에서는 정보검색시스템의 평가를 위한 다양한 테스트컬렉션을 구축해오고 있다. 1999년 TREC-8에서 질의응답시스템의 평가를 위한 테스트컬렉션의 구축을 시작한다[2].

· 본 연구는 숭실대학교의 연구지원 정책에 따라 지원을 받은 연구입니다.
† 종신회원 : 서울보건대학 전산정보처리과 교수
binsum@sh.ac.kr
†† 종신회원 : 숭실대학교 컴퓨터학부 교수
park@comp.ssu.ac.kr
논문접수 : 2004년 3월 8일
심사완료 : 2004년 8월 24일

질의응답시스템에 사용되는 검색 시스템에서는 검색된 문서들 내에 질의와 관련된 문서가 얼마나 많이 분포하고 있는 가 정확도는 얼마나 높은가 하는 것은 큰 의미가 없다. 중요한 점은 검색된 결과 내에 정답이 포함되어 있는가 하는 것이다. 이러한 관점에서 보면 가급적 많은 양을 검색해 내서 정답이 그 안에 들어있을 확률을 높이는 것이 바른 접근 방법이라 볼 수 있다. 그러나 현실적으로 주어진 환경에서 제한된 시간 내에 질의응답시스템이 분석해 낼 수 있는 처리량은 제한되어 있고, 분석 작업을 좀 더 깊고 다양하게 하여 정답을 보다 잘 찾고자 할수록 처리량은 점점 더 줄어들게 된다. 따라서 보다 깊고 정확하게 분석 작업을 수행하는 질의응답시스템일수록 보다 적게 텍스트를 뽑아주는 검색 시스템이 필요하게 된다. 이는 질의응답시스템을 위한 검색 시스템이 가급적 적은 양의 검색 결과 내에서 정답을 찾아내야 하는 요건을 갖추어야 함을 의미한다.

이에 본 논문에서는 질의응답시스템의 성능향상과 계산의 양을 줄이기 위해 질의에 가까운 문단을 찾기 위한 새로운 질의-문단 유사도 계산 방법을 제안한다. 이 방법은 질의형태에 따른 대답의 유형을 분류해서 질의와 관련 있는 문단을 구해주는 나이브 베이지안 분류기와 문장의 위치정보에 따른 계산 방법을 포함한다. 이를 통해 검색된 단락의 순위를 재조정해서 정답을 포함하고 있는 단락을 높은 순위에 위치하도록 한다[3].

2. 관련연구

2.1 문장에서의 위치정보

문장에 있어서 단어의 위치에 따른 정보검색은 MIT Sapere[4]의 연구와 각 웹 사이트의 자연어 내용을 링크 문법과 워드넷을 이용 웹 페이지의 주석을 세만틱웹의 RDF로 작성해 주는 Li[5]의 연구 등이 있다. 거의 20여 년 전에 MIT의 Katz는 의미관계 망을 사용하여 자연어 검색과 색인에 대한 연구를 하였다[6]. 문장의 구(Phrase) 색인의 기본적인 아이디어는 1987년 Fagan이 구문적인 구(자연어 분석)와 비구문적인 구(통계적 기법으로 구성된) 두 가지를 가지고 검색하는 실험을 하였다. 예를 들어 "use of an automatic text analyzer in preparation of sdi profiles"란 구는 다음과 같은 구 기술자(Descriptor)를 생성한다[7].

[automatic analyzer], [text analyzer], [preparation analyzer], [profiles preparation], [sdi profiles].

Fagan의 작업은 주로 명사구와 여기에 연결된 전치사구에 중심을 두었으며 다양한 문장들의 구성에 대한 기법들을 다루었다. 또한 통계적 기법으로 만들어진 단어 쌍과 위의 구문분석기법으로 만들어진 단어 쌍의 색인 효과를 비교하는 것으로 이루어졌다. 결과 통계적 기법으로 만들어진 단어의 쌍이 구문분석방법 보다 더 높은 정확도를 보여주었다. 그의 연구 결과 자연어처리 기법의 적용이 더 어려울 뿐 아니라 덜 효과적인 것으로 나타났다. 2001년 MIT의 Katz와 Lin은 위의 문제점을 개선한 Sapere 시스템[4]을 제안한다. 이 시스템은 변형생성문법과 X-bar 이론에 기반 한 것으로 3개의 쌍으로 이루어진 의미표현(Ternary expression)을 제안한다. 즉 Sapere에서는 다음과 같은 문장이 3단어 쌍으로 변환된다.

The big bad wolf prowled around the dark forest.
=> [wolf prowl around], [prowl around forest], [big mod wolf] [bad mod wolf], [dark mod forest]

Sapere 시스템은 앞의 경우와는 다르게 주어-동사-목적어, 형용사-명사-수식, 명사-명사-수식, 소유관계 등을 표현함으로써 좀더 많은 의미관계를 반영하고 있다. 그러나 역시 다음과 같은 문제점을 안고 있다.

The president of Russia visited the president of china => [president of China], [president of Russia], [president visit president]

위의 표현에서 과연 방문자는 누구이고 피 방문자가 누구인지 파악할 수가 없다. 즉 단일 구내의 의미는 반영하지만 문장 구조에서의 의미를 표현하지 못하는 문제가 있다. 이를 해결하기 위해서는 문장에서 각 단어의 위치정보를 반영할 수 있어야 한다.

2.2 질의응답시스템에서의 질의형태분류

본 논문에서 다루는 다른 한 분야는 질의형태분류이다. 질의형태분류를 필요로 하는 대표적인 분야는 질의응답시스템이다. 여기서는 질의형태분류를 사용하여 주어진 질의에 대한 응답 형태를 예측하는데 활용한다. 질의응답시스템은 매우 중요한 연구 분야이며 최근에 질의응답 연구의 최신 동향은 TREC을 들 수 있다[2]. Croft[8]는 질의어에 대해 상위에 위치된 문서들 안에 있는 단어들과 질의어에 같이 발생한 단어들을 찾아서 질의어를 확장하는 연구를 하였다. 이와 다르게 우리는 질의어 형식에 따라 질의어형태분류가 대답의 순위를 결정하는데 어떠한 영향을 주는지를 보여줄 것이다.

문서의 초기 집합이 검색되었을 때 전형적인 질의응답시스템은 이 문서들 안에서 대답을 추출한다. 예로, Alpha[9]는 SMART 정보검색시스템으로부터 검색된 문서들로부터 대답을 추출하는 시스템을 보여주었다. 질의어는 대답에 적당한 요소를 구성하는 질문형식으로 분류되었다. 문서는 요소를 인식하기 위해 태그 되었으며 주어진 질의어에 대해 바른 형식의 개체를 둘러싸고 있는 문장은 휴리스틱을 사용하여 순위가 정해졌다. Moldovan[10]과 Aliod[11]는 재 순위와 후 작업을 통해 최적의 문단을 찾아주는 시스템을 제안한다. Cardie[12]

는 통계기법과 언어적 지식을 결합하고 정교한 언어적 필터를 갖는 질의응답을 제안한다. TREC 8과 TREC 9의 질의응답트랙 대부분의 질의응답시스템은 일반적인 표준 TF-IDF의 변형을 사용하여 구현되었다. 문단은 휴리스틱이나 수작업으로 이루어진 정규표현(Regular Expression)을 사용하여 선택되었다.

좀더 최근에 전통적인 질의응답시스템들 중에서 일부는 정보검색의 기회를 개선하기 위해 질의어 변형을 시도하고 있다. Harabagiu[13]는 질의어 형태의 계층구조를 이용하여 질의어 변형을 하고 있다. 그러나 이런 접근들은 현저한 시스템의 개선을 보여주지는 못하고 있다. 덧붙여 AskMSR[14]은 수작업으로 질의어 변형을 사용하고 있다.

3. 위치정보와 질의형태 자동분류

본 논문에서는 일반적인 질의응답시스템의 구성요소인 표 1의 내용 중에서 4번인 응답 순위화에 관심을 갖는다. 응답 순위를 개선하기 위하여 문장에서의 위치정보 반영, 질의형태분류를 사용한 나이브 베이지안 분류기를 사용한다. 이 두 가지 방법을 사용한 새로운 질의-문단 유사도 계산 방법을 제안한다. 제안한 유사도 계산 방법을 사용하여 세계적인 정보검색대회인 TREC의 질의-데이타 순위를 조정하는 실험을 한다.

표 1 질의응답시스템의 구성요소

단계	내용
1. 질의 분석	질의 안에 있는 키워드 정의 What is autism? -> autism - 질의 형태 인식 What is 예상 대답 형구성(InsightSoft-M) {<Q; is/are; [a/an/the]; A>, <A; is/are; [a/an/the]; Q> <Q; comma; [a/an/the]; A; …
2. 문단 또는 문장의 검색	조작할 수 있는 크기의 문서 확보
3. 후보 대답 추출	extract candidate(by answer type) autism is a mental disorder that … autism, a nourishing, equivocal
4. 응답 순위화 (Ranking)	각 후보에 대해 질의에 대한 유사도 계산

3.1 문장에서의 위치정보 자동생성

전통적인 키워드검색엔진은 문서의 어떠한 이해도 하지 못하기 때문에 관련이 없는 결과가 자주 사용자에게 주어지게 된다. 이를 해결하기 위해서 개념그래프를 사용하여 자연어 문장에서 각 단어들의 위치정보를 반영할 수 있는 표현을 사용한다.

• 문장에서의 위치정보 추출과정

문장에서의 위치정보 추출 과정은 다음과 같다.

1) 웹 문서 수집
2) Tag 등 불필요한 정보 제거
3) CMU의 Link Parser를 이용한 형태소 분석
4) 구문 분석 결과 중 주어-동사-목적어, 명사-명사-수식, 전치사-명사-수식 등 관계 분석
5) 관계 분석된 결과를 개념그래프로 표현
6) 개념 그래프로 표현된 의미표현을 데이타베이스 관계로 변형하여 데이타베이스 구축

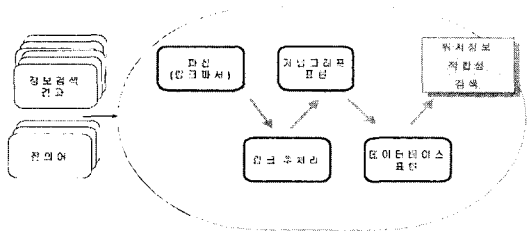

그림 1 링크파서를 사용한 위치정보추출 과정

그림 1은 위치정보추출과정을 보여주고 있다. 즉 주어진 질의어와 응답 후보는 링크파서를 통하여 파싱을 하며 후처리 과정을 거친다. 이를 개념그래프로 표현하며 키워드의 공기 가중치를 구하기 위하여 간단한 내부 데이타 표현으로 바꾼다.

• 개념그래프 변형규칙

다음은 링크파서로부터 얻은 파싱정보를 가지고 내부 데이타표현으로 변경하는 규칙을 설명한다.

1) A <--- Ax ---> B : Mod A B
2) A <--- Sx ---> B : B A [Obj]
   Agent A
3) A <--- Ox ---> B : A [Agent] B
   Obj B
4) C <--- Mx ---> A
   A <--- Jx ---> B : Mod C B

1)은 red <-- A --> rose와 같은 경우이다. 즉 앞의 단어가 뒤 단어를 형용한다. 이 경우 Mod A B와 같이 표현한다. 2)의 경우에는 주어-동사의 관계를 나타낸다. B는 동사이고 A는 주어를 표현한다. [Obj]는 목적어가 들어갈 위치이다. 3)은 동사-목적어 관계를 보여준다. 4)는 예를 들어 president <-- Mp --> of와 of <-- Js --> China가 있을 때 이 경우 Mod president China로 표현한다. 위의 경우가 아닌 Xx, Wx, Dx, Rx 등의 링크파서에서 사용하는 관계들은 문장의 직접적인 의미구조와는 상관이 없으므로 내부표현에서는 생략하도록 한다. 즉 Ax와 Jx관계로 수식어관계를 구성하고 Sx와 Ox에 의해 주어 목적어 관계를 구성해준다. 이

결과만을 사용하여 문장의 위치정보에 따른 유사도를 계산할 수 있다. 링크파서로 파싱된 결과는 앞에서 설명한 개념그래프 변형규칙을 사용하여 데이타베이스표현으로 바뀐다. 개념그래프 변형규칙에 언급된 방식대로 S와 O, J등의 링크만을 고려하여 변형을 수행하게 된다. 이와 같이 위치정보를 포함시킨 다음에 질의-문서 유사도를 구하기 위하여 대표적인 Dice Coefficient 방법(1)을 가지고 계산하게 된다.

$$\mathrm{sim}(Q,D)=\frac{2n(Q\bigcap D)}{n(Q)+n(D)} \quad (1)$$

여기에서 Q는 질의어를 D는 문서를 나타낸다. n(Q)는 질의어에서 발생한 키워드의 수를 n(D)는 문서에서 발생한 키워드의 수를 나타낸다. 이 경우 $n(Q\bigcap D)$의 의미는 질의어와 문장에서 공동으로 나타난 키워드의 수이다. 여기에서 단어에 대한 가중치는 단어가 같은 위치에(주어, 목적어) 나타났으면 1을 같은 위치가 아니더라도 같이 발생했으면 0.5를 부여한다. 다음 세 문장을 사용하여 앞의 식을 설명하기 위해 링크파싱을 하고 내부표현으로 바꾸어 위치정보기반 Dice Coefficient를 구하는 과정을 보여준다.

1) Red roses are a pretty ornament for a party.
2) Persons decorated their parties with red roses.
3) The employee decorated the roses with a red strip for a party.

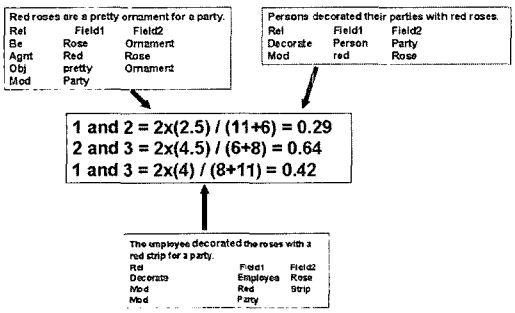

그림 2 위치정보를 반영한 유사도 계산 사례

그림 2는 위치정보에 따라 키워드 발생 값을 다르게 적용하는 Dice Coefficient를 사용한 문장들의 상호 유사도 계산 결과이다. 문장 1의 키워드 수는 11개이고 문장 2의 단어 수는 6개 이므로 1, 2문장의 키워드 수는 17이다. 또한 공동으로 발생한 단어의 수가 2.5이다(같은 위치이면 1 아니면 0.5부여). 따라서 1과 2문장의 유사도는 0.29이고 2와 3의 유사도는 0.64이다. 이 결과는 각 단어의 위치정보를 반영하였기 때문에 기존 Dice Coefficient에서 위치정보 반영 없이 단순하게 공동으로 발생한 단어에 1만을 부여한 유사도 계산에 의한 값 보다 실제 의미에 더 가까운 유사도 값을 보여준다.

3.2 나이브베이지안을 사용한 질의형태분류

가장 실제적인 베이지안 학습기 중 하나는 나이브 베이지안 학습기 이며 보통 나이브 베이지안 분류기라고 불린다. 몇몇 분야에서 뉴럴 네트워크나 의사결정트리 학습기보다 우수한 성능을 나타낸다. 이 절에서는 나이브 베이지안 분류기를 설명하고 질의형태분류에 어떻게 적용하였는지를 보여주었다.

질의어의 형태분류는 질문의 형태를 분류함으로써 질의어에 있는 키워드만을 가지고 검색하는 것이 아니라 질문의 의도를 반영한 답을 찾을 수 있게 해준다. 질의어의 형태는 다양하게 분석되고 있으며 본 논문에서는 정보검색경진대회 TREC-9의 질의응답시스템 분야 질의 693개와 뉴스그룹의 자주 대답되는 질문 문서(FAQ) 30,000여 개를 가지고 질의형태분류를 실시한다. 아래 표들은 형태분류의 결과이다. 이를 기본으로 주어진 질의형태를 분류하여 원하는 대답을 찾기 위해 분류기를 작성한다. 분류기는 뉴스그룹의 30,000여 FAQ 문서를 가지고 기계학습 분류의 대표적 방법인 나이브 베이지안을 사용하여 제작한다.

문서들 안에서 5개 질의형태분류와 관련된 문서들의 분류를 표현한다. 아래의 내용과 같이 나이브 베이지안은 각 분류에 속할 확률(2)과 문서의 각 단어에 대하여 질의형태에 속할 확률(3), 그리고 문서에 속한 단어들에 대해 각 분류에 속할 확률들을 곱하고 각 분류의 확률을 곱해서 이 값이 가장 커지는 분류가 결과가 된다(4).

$$P(v_j)=\frac{|docs_j|}{|Examples|} \quad (2)$$

$$P(w_j|v_j)=\frac{|n_k+1|}{|n+|Vocabulary||} \quad (3)$$

$$v_{nb}=\arg\max_{v_j\in V} P(v_j)\prod_{i\in positions} P(a_i|v_j) \quad (4)$$

v_{nb} : naive bayes 확률값

여기에서 $P(v_j)$는 분류 vj의 확률이고 이는 전체 예중에서 이 분류에 속한 문서의 수로 나타난다. 또한 $P(w_j|v_j)$는 vj일 때 단어 wj가 분류 그 안에 나타날 확률이다. n은 vj분류 문서의 수이며 nk는 단어 wj가 발생한 문서의 수이다. 또한 vocabulary는 학습된 데이타 안에 있는 서로다른 단어들의 수이다. vocabulary와 +1을 한 이유는 분모와 분자를 0으로 만들지 않기 위해서다. 마지막으로 식 (4)은 해당 문서가 분류 vj에 속하는지를 보여주는 확률이다. 즉 각 단어들의 분류에 속할 확률들의 곱인 $\prod_{i\in positions} P(a_i|v_j)$에 각 분류의 확률 $P(v_j)$을 곱한 것을 최대로 만드는 분류가 이 문서의 분류가 된다. (3)의 연산을 거쳐 만들어진 질의형태에 따른 단어들의 확률 값의 일부분을 다음 그림에서 보여주고 있다.

즉 what분류에서 "in"은 0.30049를 확률 값으로 갖는다.

3.3 질의-문서 유사도 계산

본 논문에서는 정보검색결과 문서들의 순위(Ranking)을 위해서 새로운 유사도 계산을 제안한다. 즉 위치정보와 질의형태분류기를 반영한 유사도 계산이다.

$$\mathrm{sim}(Q,D) = a\times\frac{2n(Q\bigcap D)}{n(Q)+n(D)} + b\times P(Q_{type})\prod_{j<positions} P(t_j|Q_{type}) \quad (5)$$

위 식은 본 논문에서 제안하는 유사도 계산식으로서 위 식에서 a와 b는 상수이다. 첫째 식은 Dice Coefficient를 사용한 문서에서 질의어와 문서에서 발생한 단어에 대한 유사도 계산이다.

$$a\times\frac{2n(Q\bigcap D)}{n(Q)+n(D)} \quad (6)$$

식 (5)에 첫 번째 (6) 경우 $n(Q\bigcap D)$에서 만약 단어가 같은 위치에(주어, 목적어) 나타났으면 1을 같은 위치가 아니더라도 같이 발생했으면 0.5를 부여한다. 둘째 식은 나이브 베이지안 분류기를 나타낸다. 즉 같은 형태의 질의일 경우에는 뒤 식에 의해 유사도 값이 가중되게 된다.

$$b\times P(Q_{type})\prod_{j<positions} P(t_j|Q_{type}) \quad (7)$$

식 (5)에 두 번째 (7)의 경우는 나이브 베이지안 분류기를 보여준다. Q_{type}은 질의형태로서 $P(t_j|Q_{type})$은 단어 t_j가 질의형태 Q_{type}에 나타날 확률이다. 이 경우 약 30,000개의 질문-답변 문서에서 학습된 각 질의형태에 있어 단어의 속할 확률 값을 단락 내에 있는 단어들로 모두 곱하고 그 질의형태의 확률을 곱하여 가장 높게 나타나는 형태가 그 분류가 된다는 내용이다. 여기에서 a와 b 값은 상수이다. 예를 들어 다음 문장을 고려해 보자. Where is Belize located? 라는 질문에 대하여 다음과 같은 문장의 유사도를 계산해 볼 수 있다.

the Belizean Government will assume responsibility for its own defense as of 1 January 1994, and announced that it had started the "immediate" withdrawal of the UK troops stationed in that country located in the *Central American*

나이브 베이지안 분류기에서 Where에 해당하는 단어들의 확률 값은 다음과 같다.

Where the .258963442524894
Where that 2.158028687707 45E-02
Where have 1.07901434385372E-02
Where he 1.39568159694123E-02
Where in .118457009488289
Where is 3.97593328876535E-02
Where it 1.02037225994863E-02
Where its 1.05555751029169E-02
Where of .111068106916247

즉 문장에서 the가 3번 its가 1번 it이 1번 in이 2번 that이 1번 of가 1번 나타난다. 이를 계산하면 유사도 = a x (2/37) + b x (0.12(where) x 0.26(the) x 3 x 0.016(its) x 1 x 0.012(it) x 1 x 0.12(in) x 2 x 0.022(that) x 1 x 0.11(of) x 1)로 표현될 수 있다. 여기에서 a와 b 값은 상수로서 임의로 정할 수 있다. (2/37)은 전체 단어 수 37개에 공동을 발생한 키워드(Belize, Located)가 같은 문장성분 위치에 발생하였으므로 2가 되어서 a x (2/37)로 표현할 수 있다.

3.4 제안 시스템의 구성

본 논문에서는 위치정보와 질의형태분류정보를 반영한 유사도 계산을 통해서 응답 단락들의 순위를 조정하여 정답이 포함된 단락이 상위에 위치하는 것을 목표로 한다. 이를 구현한 시스템의 구성을 아래 그림에서 보여주고 있다.

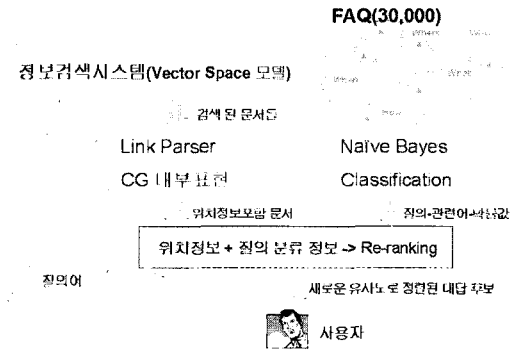

그림 3 제안시스템의 구성

그림 3은 제안시스템의 전반적인 구성을 보여주고 있다. 우선 일반적인 정보검색시스템을 통해 질의어에 대한 문서들을 찾는다. 그 다음 검색된 문서들에 대하여 일정한 크기의 단락을 만든 후(보통 3문장) 링크파서와 개념그래프 표현을 통해 문장 내의 각 단어들에 대한 위치정보를 구한다. 그리고 나이브 베이지안 분류기를 사용하여 질의에 대한 관련을 확률 값으로 구한다. 결과 사용자의 질의에 대하여 유사도가 높은 순으로 응답결과를 보여준다.

4. 실험 및 평가

과거의 정보검색과 관련된 테스트 컬렉션들은 매우 적은 수의 문서들을 대상으로 구축된 소규모 테스트 컬렉션들이다. 따라서 산업체에서는 이러한 소규모 테스트 컬렉션을 사용한 정보 검색연구 결과들을 신뢰하지 않

는 경향을 보여 왔다. 그러나, 미국의 NIST(National Institute of Standards and Technology)의 후원으로 1992년에 처음으로 개최된 학술 대회 TREC(Text REtrieval Conference)에서 1백만 건을 초과하는 문서들을 대상으로 대용량 테스트 컬렉션의 구축을 시작하였으며, 이후 매년 테스트 컬렉션에 포함되는 문서들의 수를 증가시키고 있다. 본 논문에서는 실험을 위해서 TREC-9의 693개의 질의어를 가지고 실험한다. 대상 데이타는 이 693개의 질의어에 대해 TREC에 참가한 28개의 팀이 제출한 정답 50바이트와 250바이트 크기의 데이타이다. 각 질의 당 평균 304개의 데이타이며 이중 정답을 포함하는 데이타는 약 10%에 해당하는 25개 정도이다. 전체 데이타의 개수는 21만 여개이며 질의 693개에 대하여 이 21만개의 데이타들의 유사도를 구하고 순위를 계산하였다.

이에 대한 평가는 TREC에서 사용하는 평균상호순위(MRR, Mean Reciprocal Rank, 이하 MRR로 표기)를 사용한다. 이 값은 정답이 1위에 위치되었을 때 1, 2위에는 1/2, 3위에는 1/3 값을 취하며 5위 이후에 위치했을 때는 0을 값으로 취한다. 전체 질의어의 상호순위 값의 평균이 평균상호순위이다.

표 2는 TREC-9의 50바이트 제한 제출에서의 순위를 가지고 실험결과를 비교한 것이다. MRR 값이 5위를 차지한 IBM의 값과 같고 6위의 퀸스대학 보다 0.01정도 높은 수치를 보이고 있다. 앞의 방법들이 수작업을 통한 패턴 매칭 등을 사용하는 것에 비하여 자동학습기법을 사용한 본 제안이 의미를 갖는다고 볼 수 있다.

다음 표는 각 질의 형태에 따라 질의형태분류기를 적용한 결과와 위치정보를 반영한 결과 MRR 값이 어떻게 변하는지를 보여주고 있다. 표를 통해서 대부분의 질의형태가 분류기를 통해 MRR 값이 평균값 이상으로 상승했음을 알 수 있다. 즉 질의형태분류기의 적용결과가 질의형태와 관련 있는 질의들에게 효과적임을 나타내고 있다.

표 2 TREC-9 시스템들과의 비교(MRR)

실험유형	제안 시스템	Queens 대학 (6)	IBM (5)	IBM (4)
키워드만 적용	0.13	0.28	0.29	0.32
질의형태분류기 적용	0.27			
질의형태분류기 + 위치정보	0.29			

표 3 질의 형태에 따른 MRR 개선 값 비교

실험유형	What	When	Where	Who	How
키워드만 적용	0.13	0.12	0.12	0.14	0.1
질의형태분류기 적용	0.31	0.29	0.33	0.36	0.26
질의형태분류기 + 위치정보	0.32	0.29	0.35	0.36	0.27

다음 실험은 질의형태를 좀더 세분화하여 질의형태세분화가 단락의 재순위에 어떠한 영향을 주는지에 관한 것이다. 이를 위하여 우선 30,000개의 FAQ 문서에서 50회 이상 발생하는 질의형태를 가지고 분류한다. 따라서 표 4와 같이 22개 질의형태 세부 분류기를 사용하여 유사도 계산을 수행한 경우 5개 분류기를 사용한 경우보다 더 좋은 결과를 얻지 못했다.

표 4 질의형태 세부 분류의 평균상호순위 비교

분류	5개 질의형태분류	22개 질의형태분류
평균상호순위	0.29	0.27

나이브 베이지안 분류기의 실제 분류(Classification) 성능을 알아보기 위하여 반대로 높은 확률 값을 갖는 분류의 질의형태와 원래 질문의 형태가 같은지를 비교하는 실험을 한다. 693개의 질문에 대하여 분류기의 성능을 측정하기 위하여 위와 같은 실험을 시행한다. 결과 질의어 693개 중에 5개의 질의형태 분류로 되어있는 질의어 581개에 대하여 356개를 바르게 예상하였고 정확도는 61.3%를 기록한다.

표 5 분류기 성능평가를 위한 분류 정확도

전체	Wh 형태의 질의 수	분류에 성공한 개수	정확도
693	581	356	61.3%

다음 실험으로 전체 질의 693개 중에서 상위 5위 안에 정답 포함 데이타를 위치시켰던 개수를 측정한다. 결과 382개였으며 전체에 대해 55.1%의 정확도를 보여주었다.

다음은 이항분포를 따르는 사건을 가정하여 질의 당 304개의 답변에 대하여 정답이 25개 포함되어있을 때 상위 5개에 정답이 있을 확률을 구해보았다. 이항분포이므로 이 사건의 평균 확률은 0.41이고 표준편차는 0.6이다. 일반적인 확률로 보면 평균 41%의 값을 갖는다.

$$\mu = np = 5 \times (\frac{25}{304}) = 0.41$$

$$\sigma = \sqrt{npq} = \sqrt{5 \times 0.08 \times 0.92} = 0.6$$

본 시스템에서 측정한 값은 55.1%이다. 따라서 이항분포를 갖는 일반적인 사건의 평균 확률 값보다 훨씬 높은 것을 알 수 있다. 이 값은 TREC-9의 50바이트 제출 정답수행의 평균 정확도 35%보다 높고 250바이트 데이타의 평균 54% 보다 높은 수치이다. 실험한 데이타가 약 40%의 50바이트 데이타와 60% 250바이트 데이타를 포함하고 있는 경우를 고려하면 TREC에 정답을

제출한 상위 20개 팀의 평균 정확도 보다 높게 나타난 것으로 볼 수 있다.

표 6 전체 질의에 있어 5위안에 정답을 위치시킨 정확도

전체 질문 개수	5위안에 정답이 포함된 개수	정확도
693	382(50, 250바이트 포함)	55.1%

표 7은 TREC-9에 참가한 20개 팀이 제출한 질의에 대한 상위 5개 50바이트, 250바이트 자료의 통계이다. 전체가 21만개 정도이며 이중 정답을 포함하는 경우가 18000개로 10%정도 차지하는 것을 알 수 있다. 이를 693개의 질의어를 감안하면 질의 당 평균 304개의 데이타가 제출된 것이며 이중 정답을 포함한 데이타는 25개 정도로 볼 수 있다. 또한 TREC-9의 상위 20개 팀의 250바이트 제한에서 답을 발견한 평균은 377개 정도로 약 54%의 정확도를 보이고 50바이트에서는 245개로 정확도 35%를 나타낸다.

추가로 포항공대에서 개발한 질의응답시스템의 유사도 순위 단락들을 가지고 실험하였다. 이 실험은 TREC 9에 실제로 참가하였던 결과물로서 492개의 질의에 대한 응답 문단을 가지고 있으며 내용은 질의 각각에 대하여 100개의 유사도가 높은 순으로 나열된 문단들로 이루어져 있다.(49,200개) 이 결과물에 본 논문의 유사도 계산 방법을 적용하여 정답의 상위(5위) 문단 포함확률을 계산하였다.

결과는 3개의 문장으로 이루어진 문단에 대한 새로운 유사도 방법에 순위 개선이 0.27에서 0.35로 나타났으며 250바이트로 정답 문단을 추출하여 상위 5위 안에 정답이 속하였는지를 측정한 결과는 기존 시스템 0.33에 비해 조금 낮은 0.3을 기록하였다. 이 결과는 기존의 시스템이 정규표현 등 여러 가지 기법을 사용하여 복잡한 과정으로 정답을 추출하는데 비하여 본 시스템은 유사도 계산 만을 사용하여 질의어 키워드 중심으로 250바이트를 추출하였기 때문이다.

5. 결 론

본 논문에서는 질의응답시스템의 성능을 개선하기 위한 방법들을 제안하였다. 대량의 문서집합 속에서 사용자가 원하는 정보를 검색한다는 점에서 질의응답 시스템은 정보 검색 시스템과 유사한 특징이 있다. 그러나 일반적으로 정보 검색 시스템이 사용자의 질의와 관련된 문서들을 찾는 데 반해 질의응답시스템은 질의에 대한 정확한 답을 찾아야 한다는 점에서 일반적인 정보 검색 시스템보다 더욱 정밀한 검색 작업이 요구된다. 즉, 정보 검색 시스템에서 사용되는 색인 가능한 기본적인 정보 이외에도 색인 할 수 없는 다양한 구문 정보 혹은 의미 정보들을 사용하여 정답임을 판별해 내는 분석 작업을 수행한다. 본 논문에서는 새로운 질의-문서 유사도 계산을 사용하여 단락의 순위를 조정하여 질의응답시스템의 성능을 향상하기 위한 방법을 제안하였다.

첫째로 위치정보의 반영이다. 카네기멜론대학의 링크 파서와 개념그래프를 사용하여 문서의 자연어표현에 있는 주어-목적어, 수식어-명사 등의 관계를 추출하였다. 이렇게 함으로써 문장에 의미구조를 정보검색 결과에 반영할 수 있었다. 다른 제안 방법은 질의형태분류이다. 기존의 질의 분석이 거의 수작업을 통한 패턴을 추출하는 방법을 사용하는데 비하여 우리의 방법은 자동화된 기계학습을 사용하였다. 뉴스그룹에서 발췌한 FAQ (Frequently Asked Questions) 30,000여 개를 가지고 질의형태에 따른 질의어 분류를 하였다. 이를 정보검색에 반영하기 위하여 나이브 베이지안을 사용하여 분류기를 작성하였다.

제안한 방법에 의해 TREC-9의 693개 질의어에 대한 참가한 28개 팀들이 제출한 21만여 개의 50바이트, 250바이트 데이타에 대해 상호순위를 실험하였으며 평균상호순위가 TREC-9에 참가한 팀의 MRR 값 중에서 4위와 5위 사이인 0.29를 기록하였다. 질의어를 세분화하여 22개의 형태로 분류하는 실험도 실시하였는데 질의어를 5개로 분류한 것에 비하여 별 차이가 없는 결과를 보였다. 이것은 베이지안 분류기가 각 단어에 대하여 독립적인 확률을 갖는 것 때문으로 추정된다. 분류성능에 대한 실험을 통해서 단락의 단어들에 분류기 확률 값을 적용했을 때 질의에 형태를 맞추는지를 실험하였다. 결과로

표 7 TREC-9 28개 팀 78개 수행에 대한 통계

전체단락 수	답포함 단락 수	1질의 당 평균 단락 수	1질의 당 답포함 단락 수	TREC-9의 250바이트 제한에서 답포함 단락 수	TREC 9의 50바이트 제한에서 답포함 단락 수
210,948	18,005	304	25	376.95/54%	245.2/35%

표 8 포항공대의 TREC-9 참가 데이타

기존 유사도 순위에 MRR	새로운 유사도 방법에 따른 MRR	기존 250바이트 크기 문단 MRR	새로운 유사도 방법 250바이트 MRR
0.27	0.35	0.33	0.30

자동학습을 통해 얻은 나이브 베이지안 분류기가 61.3% 정도의 분류성능을 나타내었다. 또한 정답을 찾아낸 정확도가 질의 693개 중에서 382개로서 TREC-9에 참가한 상위 20개 팀의 50바이트, 250바이트 제안 평균보다 높은 약 55.1%정도로 나타났다.

질의형태분류를 통한 나이브 베이지안 분류기의 사용은 질의 분석에서 자동화된 기계학습방법을 사용한 것이다. 또한 개념그래프를 사용하여 문장에서 문법적인 성분으로부터 위치정보를 추출하는 방법도 자동화된 링크파서를 사용한 자연어 처리기법을 사용하였다. 기존의 방법들이 수작업을 통한 예상 답변들의 패턴을 구하여 접근하는 방법인데 비해서 본 논문에서는 자연어처리 및 기계학습 방법을 통한 자동화에 초점을 맞추었으며 그 의의를 갖는다.

참 고 문 헌

[1] 이경순, 김재호, 최기선, "질의응답 시스템의 성능 평가를 위한 테스트컬렉션 구축", 제12회 한글 및 한국어 정보처리 학술대회, pp. 190-197, 2000.

[2] Voorhees, E. and Harmon, D., "Overview of the TREC 2001 Question Answering Track," TREC-10 Proceedings, 2001.

[3] 이영신, 황영숙, 임해창, "질의응답 시스템을 위한 가변 길이 단락 검색", 제14회 한글과 한국어정보처리 학술대회, pp. 259-266, 2002.

[4] Lin, J., "Indexing and Retrieving Natural Language Using Ternary Expression," Master's Thesis, Massachusetts Institute of Technology, 2001.

[5] Li, J. and Yu, Z., "Learning to Generate CGs from Domain Specific Sentences," The Proceedings of the 9th International Conference on Conceptual Structures, 2001.

[6] Katz, Boris and Winston, Patric H., "A two-way natural language interface," In proceedings of the *European Conference on Integrated Interactive* Computing Systems, 1982.

[7] Fagan, Joel L., "Experiments in Automatic Phrase Indexing for Document Retrieval," Ph.D thesis, Cornell University, 1987.

[8] Xu, J. and Croft, W. B., "Improving the effectiveness of information retrieval with local context analysis," ACM Transaction on Information Systems, vol. 18, No.1, pp.79-112, 2000.

[9] Alpha, S. Dixon, P. Liao, C., "Oracle at TREC 10," TREC-10 Proceedings, 2001.

[10] Moldovan, D., "A tool for surfing the answer net," TREC-8 Proceedings, 1999.

[11] Aliod, D. and Berri, J., "A real world implementation of answer extraction," In Proceedings of the 9th International Workshop on Database and Expert Systems, 1998.

[12] Cardie, C. and Pierce, D., "Examining the role of statistical and linguistic knowledge sources in a general-knowledge question answering system," ANLP-2000, 2000.

[13] Harabagiu, S. M., "Experiments with open-domain textual question-answering," COLING-2000, 2000.

[14] Hovy, E.H., "Question Answering in Webclopidia," TREC-9 Proceedings, 2000.

[15] Strzalkowski, Tomek., "Natural Language Information Retrieval," TREC-5 Proceedings, 1996.

김 명 관

1985년 숭실대학교 전자계산학과(학사) 1987년 숭실대학교 대학원 컴퓨터학과(석사). 2004년 숭실대학교 대학원 컴퓨터학과(박사). 1989년~1993년 한국전자통신연구원 인공지능연구실 연구원. 1993년~현재 서울보건대학 컴퓨터정보과 부교수. 관심분야는 기계학습, 에이전트, 정보검색, 자연어처리 등

박 영 택

1978년 서울대학교 전자공학과(학사) 1980년 KAIST 전자계산학과(석사). 1992년 Illinois at Urbana-Champaign 컴퓨터과학과 박사. 1981년~현재 숭실대학교 컴퓨터학부 교수. 관심분야는 지능에이전트, 웹에이전트, 모바일에이전트, 기계학습 등