빈발질의를 추천하는 질의 응답 시스템

안찬민*, 최범기*, 이주홍*
*인하대학교 IT 대학 컴퓨터정보공학부

e-mail:ahnch1@datamining.inha.ac.kr*
bumghichoi@yahoo.co.kr*
juhong@inha.ac.kr*

Question Answering System with Recommending FAQ

Chan-Min Ahn*, Bumghi Choi*, Ju-Hong Lee*
*Dept of Computer Science & Information Technology, Inha University

요 약

질의 응답 시스템은 사용자가 입력한 질의에 대한 답변 문장들을 보여주는 시스템이다. 대부분의 기존의 연구는 사용자의 질의문에 대해서 가장 적합한 문장들을 찾는 방법을 제안하고 있다. 그러나 질의문에 사용되는 단어들은 근본적으로 애매모호성을 포함하고 있기 때문에, 시스템이 사용자의 정확한 질의 의도를 파악하여 가장 적합한 문장들을 찾는 것은 불가능하다. 이러한 근본적인 문제를 개선하기 위해서 여러가지 연구들이 수행되었다. 본 논문에서는 이러한 문제점을 해결하기 위한 방법으로서 시스템에서 답변이 준비된 빈발 질의(FAQ)들 중에서 사용자의 질의를 함의하는 것들을 추천하여 사용자가 자신의 질의 의도에 따라 정확한 답변을 효과적으로 찾도록 도와주는 방법을 제안한다.

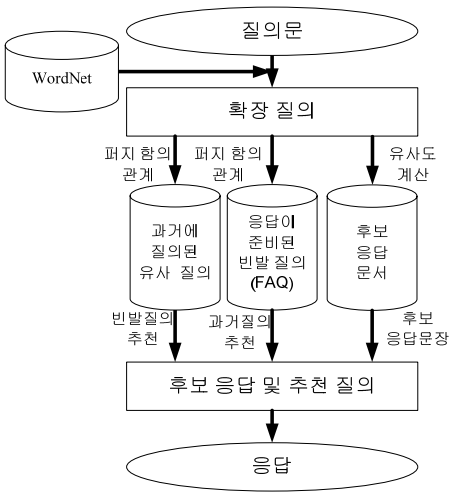
1. 서론

질의 응답 시스템(Question answering system)은 자연언어 형식의 사용자 질의를 받으면 문서를 검색하여 질의에 연관된 응답 문장을 찾아 제공하는 시스템이다. 대부분의 기존 연구들은 단어가 근본적으로 포함하고 있는 애매모호성을 극복하지 못한다. 단어가 가지고 있는 의미는 여러 종류가 있고 그 단어가 쓰인 문장에 따라 다른 의미로 사용된다. 따라서 같은 단어를 사용한 질의문이라도 사용자가 의도하는 응답은 다를 수 있다. 그러나 질의 응답 시스템은 질의문의 길이가 짧기 때문에 시스템이 사용자의 정확한 의도를 파악하는 것은 불가능하다.

본 논문에서는 시스템이 사용자의 질의의 의도를 정확히 분석하는 것은 사실상 불가능하다는 전제하에 그에 대한 대안으로서 사용자의 질의를 포괄적으로 함의할 수 있는 준비된 질의들을 검색하여 사용자에게 질의와 그에 대한 답변들을 보여 줌으로써 사용자가 스스로 적합한 답변을 손쉽게 검색할 수 있도록 도와주는 시스템을 제안한다. 이 때 준비된 질의에는 답변이 준비된 질의 (FAQ)와 결과에 대한 만족도가 높은 다른 사용자의 질의와 그에 대한 답변 결과들이 포함된다. 준비된 질의에 대한 검색은 시스템의 응답 결과에서 사용자가 원하는 답변이 없거나 부족하다고 판단되는 경우에 수행된다.

사용자의 질의를 포괄적으로 함의하는 준비된 질의를 찾는 방법으로서 퍼지 함의 연산자 기법을 사용하였다.

그림 1 은 본 논문에서 제안하는 시스템의 구성요소들과 그 관계를 나타낸다.



(그림 1) 제안하는 시스템의 구성요소 및 관계

시스템은 입력된 질의문에 대한 응답 문장을 찾기 위해 질의문을 확장한다. 확장된 질의는 후보 응답 문서에서 후보 응답 문장을 찾는 작업을 수행한다. 그리고 입력된 질의문을 함의하는 추천 질의문을 미리 준비된 빈발 질의 집합과 과거 시스템에 질의된

질의 집합에서 찾는 작업을 수행한다. 만일 시스템이 찾아낸 후보 응답 문장이 사용자의 의도를 만족하지 못한다면 추천된 질의를 이용하여 사용자의 의도를 만족하는 응답을 얻을 수 있도록 돕는다.

본 논문의 구성은 다음과 같다. 2 절에서 이전까지 수행된 질의 응답 시스템에 관한 관련 연구를 보인다. 3 절에서 본 논문에서 제안하는 방법을 소개한다. 4 절에서 결론을 맺는다.

2. 관련 연구

질의 응답 시스템에 대한 정형화된 구조는 [1]에서 정의되었다. Text RErtieval Conference (TREC)에서는 Question answering track 을 만들어서 1999 년부터 질의 응답에 관련된 연구를 진행하고 있다[2].

질의 응답시스템은 크게 질의의 종류에 따라 사실 질의(Factoid question)와 정의 질의(Definitional Question)로 구분된다. 사실 질의는 사용자가 대상에 대해 어느 정도 알고 있으나 그에 대한 구체적인 정보가 필요한 경우에 사용된다. 예를 들면 “When did the Vietnam War begin?”과 같이 베트남 전쟁을 어느 정도 알고 있는 사람은 전쟁 발발 시기와 같은 구체적인 정보를 질의할 수 있다.

응답을 효율적으로 검색하기 위해 사실 질의를 예측되는 응답 형태에 따라 질의를 분류하는 방법(Question classification)에 관한 연구가 진행되었다. 이 방법은 검색 범위를 분류 카테고리로 한정하여 검색하기 때문에 연산 비용과 정보의 정확도가 향상된다[3,4].

정의 질의는 사용자가 대상에 대해 정확한 정보를 알지 못할 경우 정의 문장 형태의 응답을 원하는 경우 사용한다. 질의문에는 찾고자 하는 대상을 표현하는 단어만 포함되어 있다. 예를 들면 “What is the Vietnam War?”와 같은 질문은 한 두 개의 응답 문장으로는 정확한 답변을 할 수 없다. 따라서 미리 정의된 정의 형태의 구문과 유사한 문장 구조를 가지면서 질의문과 연관된 문장을 정의된 응답 문장으로 제공하는 방법이 연구되었다[5,7].

Cui 는 질의문을 구성하는 질의어 사이의 관계를 고려한 질의 응답 방법을 제안하였다[5]. 기존의 방법은 의미적으로는 같더라도 문장을 구성하는 단어들의 관계가 다르면 유사하지 않은 문장으로 판단하는 문제를 갖는다. 이 문제점을 극복하기 위해 Cui 는 질의문을 구성하는 질의어 사이의 관계 매칭에 퍼지 관계를 적용하였다. 이 방법은 질의어 자체만을 고려하지 않고 질의어들 사이의 관계를 고려함으로써 단어가 갖는 모호성을 극복하고자 하였다. 그러나 이 방법은 문장 구조의 유사성에 초점을 맞추었을 뿐 사용자의 질의 의도는 고려하지 않았다.

Prager 는 정의형 질의를 몇몇 보조 사실 질의로 분할하고, 보조 사실 질의들의 응답을 정의형 질의의 응답으로 조합하는 방법을 제안하였다[6]. 보조 사실 질의는 정의 질의의 종류(사람,조직,사물)에 따라 정해진다. 이 방법은 질의에 대한 응답을 찾기 위해 사용자가 직접 보조질의를 입력해야 하는 단점을 갖는다.

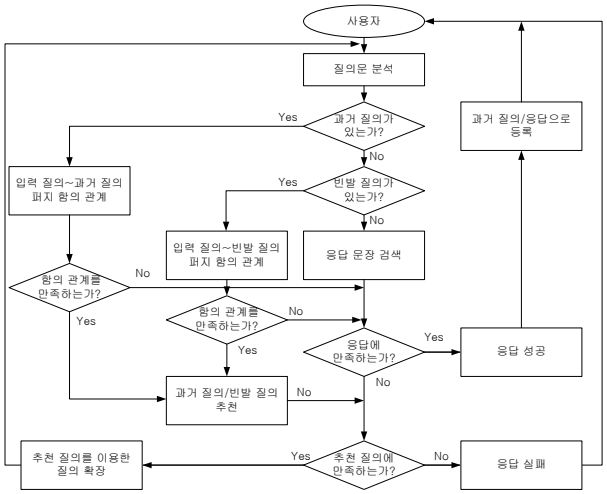
Kor 는 외부 정보로부터 Human interest model 을 구축하는 방법을 제안하였다[7]. 이 방법은 질의에 대한 사실 정보와 흥미 정보를 함께 고려하였다. 예를 들면 “George foreman”에 대한 사실 정보와 흥미를 유발하는 정보를 외부 정보로부터 입수하여 질의와 함께 제공하는 방법을 제안하였다.

위 방법들은 질의문의 애매모호성을 극복하기 위해 문장 구조를 분석하거나 추가 질의를 통해 응답을 찾기 위한 범위를 확장하거나 이미 정의된 외부 정보를 이용하였다.

3. 추천 질의 응답 시스템

3.1 시스템의 구성

본 논문에서 제안하는 추천 질의 응답 시스템의 흐름은 그림 2 와 같다.

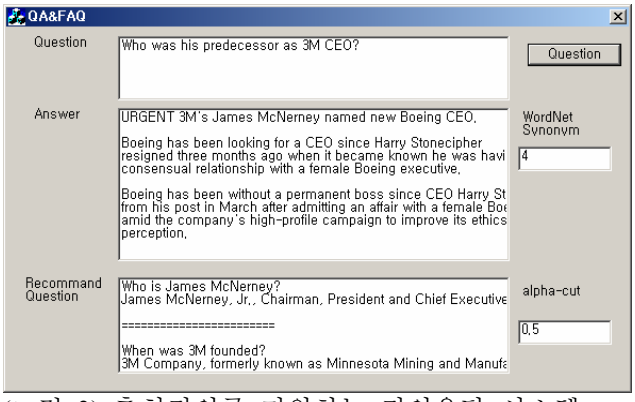


(그림 2) 추천 질의 응답 시스템의 흐름

사용자가 질의문을 입력하면 시스템은 입력된 질의문을 분석한다. 만일 과거에 질의된 질의문 혹은 응답이 준비된 빈발 질의문 중에서 입력된 질의문을 함의하는 질의가 있다면 추천 질의로 제공할 수 있도록 한다. 이는 질의에 대한 응답 문장이 사용자가 의도하는 정보를 나타내지 못할 수도 있기 때문에 응답 문장과 함께 추천 질의도 제공하기 위함이다. 질의문간의 포함 관계는 퍼지 함의 관계로 얻는다. 시스템은 입력된 질의문을 포함하는 의미를 갖는 준비된 질의문들을 검색한다.

그 후 질의에 대한 후보 응답 문장을 사용자에게 제공한다. 사용자가 응답 문장에 만족하면 응답 문장을 찾는데 성공하였음을 사용자에게 알린다. 그리고 성공한 질의 응답은 차후 질의 응답에 활용할 수 있도록 과거 질의에 추가한다. 질의문에 대한 적합한 응답 문장의 기준은 질의문과 응답 문장 사이의 유사도로 계산된다. 유사도를 계산하는 방법은 벡터 모델, 확률 모델 등 여러 가지가 있으나 일반적으로 BM25 가 가장 좋은 성능을 보이는 방법으로 인정하고 있다[8]. 본 논문에서는 문장간의 유사도를 확률 모델의 일종인 BM25 를 이용하여 계산한다[9].

만일 사용자가 응답 문장에 만족하지 못하면 질의문에 함의된 추천 질의를 사용자에게 제공한다. 추천 질의에 만족하면 사용자는 질의를 확장한 후 시스템에게 다시 질의할 수 있다. 그러나 추천 질의에도 사용자를 만족시키는 응답 문장이 없는 경우 응답 문장을 찾는데 실패하였음을 사용자에게 알린다. 그림 3 은 본 논문에서 제안하는 추천질의를 지원하는 질의응답 시스템을 나타낸다.



(그림 3) 추천질의를 지원하는 질의응답 시스템

사용자는 질의문을 입력하고 파라미터를 설정한 후 질의를 시작하면 후보 응답 문서에서 유사도가 높은 문장들을 출력한다. 동시에 응답이 미리 준비된 빈발 질의들 중에서 질의문을 함의하는 질의와 응답 문장을 같이 보여준다. 따라서 사용자는 응답 문장이 의도하지 않은 내용일 경우, 추천된 빈발 질의와 응답을 참조하여 의도하는 내용으로 질의를 확장할 수 있다.

3.2 퍼지 함의 관계

사용자가 질의를 통해 응답을 얻지 못했거나 얻어진 응답이 만족스럽지 못할 경우 질의를 확장하면 좀 더 유용한 응답을 얻을 수 있다. 본 논문에서는 질의를 자동으로 확장할 수 있도록 입력된 질의와 준비된 질의들 사이에 퍼지 함의 관계를 적용한다. 준비된 질의들 중에서 입력된 질의보다 넓은 의미를 포함하는 질의를 사용자에게 하위 질의로서 추천하면 사용자가 원하는 정보를 얻기 위해 질의를 확장하는 데 유용하게 활용할 수 있다.

질의문 Q_i 가 Q_j 에 포함되는 정도는 다음 식과 같이 정의된다[10].

$$Q_i \xrightarrow[\alpha]{} Q_j = (R_\alpha^T \Delta R)_{ij} = \frac{1}{|Q_{i\alpha}|}\sum_{t\in Q_{i\alpha}} (R_{it}^T \rightarrow R_{tj}) \quad (1)$$

Q 는 질의문이다. R 은 후보 응답 문장과 질의문 사이의 퍼지값으로 구성된 m x n 행렬이다. t 는 α-cut 을 만족하는 Q 를 구성하는 퍼지값이다. $Q_{i\alpha}$ 는 α-cut 보다 큰 Q_i 값이다. α 는 퍼지 관계 연산자의 α-cut 값이다. 후보 응답 문장과 질의문 사이의 유사도는 BM25 를 이용한다[9]. BM25 는 그림 4 와 같다.

후보 응답 문장 C 와 질의어 Q 사이의 유사도는 식 (2)로 계산한다. 그리고 퍼지 관계 행렬에 적용하기 위해 유사도 값은 정규화를 통해 0~1 사이의 퍼지값으로 변환하였다.

$$sim_{BM}(Q,D) = \sum_{t\in T} idf \cdot W \quad (2)$$

Q: 질의 문장
D: 후보 응답 문장
T: Q 와 D 에 포함된 모든 단어 t 의 집합

$$idf = \log \frac{N - n + 0.5}{n + 0.5}$$

N: Text collection 의 문서의 수
n: term 이 나타난 문서의 수

$$W = \frac{tf\,(k_1+1)}{k_1\left((1-b)+b\times\frac{dl}{avdl}\right)+tf}$$

$K_1 = 2,\ b = 0.75$
tf : term frequency
dl : 문서의 길이=term 의 수
$avdl$: text collection 의 평균 문서의 길이

(그림 4) BM25 Scoring Model

본 논문에서는 입력된 질의를 $\tilde{Q} = \{\tilde{q}_1, \tilde{q}_2 ..., \tilde{q}_n\}$ 으로 정의하였다. $\tilde{q}_1,...,\tilde{q}_n$ 은 질의문을 구성하는 단어들이다. 응답이 준비된 빈발 질의문 $Q_1, Q_2, ..., Q_m$ 의 집합을 $Q_{FAQ=}\{Q_1, Q_2, ..., Q_m\}$ 로 정의한다. $C_1, C_2, ..., C_n$ 은 후보 응답 문서를 구성하는 문장들이다. 각각의 후보 응답 문장은 $C_n = \{c_1, c_2, ..., c_k\}$ 로 정의한다. $c_1,...,c_k$ 은 후보 응답 문장을 구성하는 질의어들 중에서 불용어를 뺀 어간 형태의 단어들이다.

입력된 질의문과 추천 질의 문장 사이의 유사도로 구성된 행렬의 예는 표 1 과 같이 나타난다.

<표 1> 질의문과 후보 응답 문장 사이의 퍼지 관계 행렬

	$\tilde{Q}$	Q_1	…	Q_n
C_1	0.9	0.1	…	0
C_2	0.72	1	…	0.45
…	…	…	…	…
C_m	0.95	0	…	0.2

(a)R

	$\tilde{Q}$	Q_1	…	Q_n
$\tilde{Q}$	0.77	0.49	…	0.35
Q_1	0.78	0.93	…	0.75
…	…	…	…	…
Q_n	0.8	0.9	…	0.75

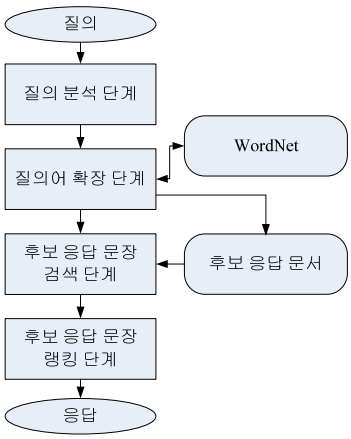
(b)$R^T \Delta R$

표 1-b 는 표 1-a 의 행렬 R 과 R 의 전치행렬을 식 (1)을 통해 계산한 결과이다. 표 1-b 의 값은 열의 질의문 Q 가 행의 질의문 Q 에 함의되는 정도를 나타내는 퍼지값이다. 예를 들면 $Q_2 \rightarrow Q_1$ 은 $(R^T\ \Delta\ R)_{21}$=0.78 이다.

3.3. 질의에 대한 응답 검색 모듈

질의에 대한 응답 검색 모듈의 흐름은 그림 5 와 같다. 사용자가 질의를 입력하게 되면 시스템은 질의를

구성하는 단어들을 분석하여 필요한 정보들을 얻는다. 대체로 관사나 전치사, 조사 등과 같은 단어들은 출현 빈도가 높지만 상대적으로 중요한 정보는 포함하지 않는다. 이러한 단어들을 불용어(Stopword)라고 한다. 또한 단어의 복수 형태나 과거 형태도 정보 검색에서는 큰 의미를 갖지 않는다. 단어에서 변화하지 않는 부분을 어간(Stem)이라고 한다. 질의 응답 시스템에 사용되는 문장들은 불용어를 제거하고 어간만을 남겨서 정보를 정리하는 작업이 필요하다.



(그림 5) 질의에 대한 응답 검색 모듈의 흐름

시스템의 첫번째 단계인 질의 분석 단계에서는 이 작업을 수행한다. 질의 분석을 위해 GATE 언어 분석 툴을 이용하였다[12]. 이 툴을 이용해서 문장에 포함된 불용어를 제거하고 어간만을 얻는다. 예를 들면 “Where is the company based?”라는 질의가 입력된 경우, 질의 분석 단계를 거치면 Where, company, base 를 얻게 된다. 그러나 대체로 문장의 길이가 짧기 때문에 얻어진 질의어만으로는 사용자가 원하는 정보를 충분히 얻지 못할 가능성이 높다.

이를 위해 두번째 단계에서는 질의어를 확장한다. 본 논문에서 제안하는 시스템에서는 질의어를 확장하기 위해 동의어 사전인 WordNet 을 이용하였다[11]. 예를 들면 단어 base 를 WordNet 으로 확장하면 establish, base, ground, found 로 확장할 수 있다.

세번째 단계에서는 질의의 응답이라고 판단되는 후보 응답 문장들을 얻는다. 후보 응답 문장은 질의에 대해 전문가들이 관련성이 높다고 분류한 후보 응답 문서를 구성하는 문장들이다.

네번째 단계에서는 얻어진 후보 응답 문장들은 유사도가 높은 순서대로 순위화한다. 유사도가 높을수록 질의에 대해 적합한 응답이다. 질의 응답 시스템은 최종적으로 높은 유사도를 가진 상위 후보 응답 문장들을 응답으로서 사용자에게 제공한다.

4. 결론

본 논문에서는 사용자가 입력한 질의에 대해 후보 응답 문장과 응답이 준비된 빈발 질의를 추천하는 새로운 질의 응답 시스템을 제안하였다. 단어의 모호성으로 인해 사용자가 질의하는 대상에 대해 의도하는 응답을 시스템이 파악할 수 없다는 점에 착안하여, 미리 응답이 준비된 빈발 질의들 중에서 질의의 내용을 함의하는 질의를 추천하여 사용자의 의도에 부합하도록 질의를 확장한다.

향후 연구에서는 두 단어 이상으로 구성된 단어구 처리와 같이 질의 분석 범위를 확장하여 보다 일반화된 질의 응답 시스템을 구성할 예정이다.

참고문헌

[1] L. Hirschman, R. Gaizauskas, "Nuatural Language Question Answering: The View from Here", Natural Language Engineering 7 (4), Cambridge University Press, 2001 pp 275-300

[2] Text Retrieval Conference Question Answering Track, http://trec.nist.gov/data/qamain.html

[3] D. Zhang, W.S. Lee, "Question Classification using Support Vector Machines", ACM SIGIR, 2003, pp 26-32

[4] X. Li, D. Roth, "Learning Question Classifiers: The Role of Semantic Information", Natural Language Engineering 12 (3), Cambridge University Press, 2005, pp 229-249

[5] J. Prager, J. Chu-Carroll, K. Czuba, C. Welty, A. Ittycheriah, R. Mahindru, “IBM’s PIQUANT in TREC 2003”, In Proceedings of the 12th Text Retrieval Conference, 2003, pp 283-292

[6] H. Cui, M.Y. Kan, T.S. Chua, "Generic Soft Pattern Models for Definitional Question Answering", ACM SIGIR 2005, pp 384-391

[7] K.W. Kor, T.S. Chua, “Interesting Nuggets and Their Impact on Definitional Question Answering”, ACM SIGIR 2007, pp 335-342

[8] L. Wang, W. Fan, W. Xi, E.A. Fox, “Can We Get a Better Retrieval Function From Machine?”, In Proceedings of the 13th Text Retrieval Conference, 2004

[9] Robertson S.E., Walker S., Jones S., Hancock-Beaulieu M.M., Gatford M., “Okapi at TREC-3”, In Proceedings of the 3rd Text Retrieval Conference, 1995, pp 109-126

[10] B.G. Choi, J.H. Lee, S. Park, “Dynamic Construction of Category Hierarchy Using Fuzzy Relational Products”, IDEAL 2003, LNCS 2690, pp 296-302

[11] WordNet - a lexical database for the English language, Princeton University Cognitive Science Laboratory, http://wordnet.princeton.edu/

[12] GATE - A General Architecture for Text Engineering, http://gate.ac.uk/