

Lexico-Semantic Pattern을 이용한 오픈 도메인 질의 응답 시스템*

이승우 정한민 광병관 김동석 차정원 안주희 이근배 김학수* 김정선* 서정연**
포항공과대학교 컴퓨터공학과
다이렉트닷컴*, 서강대학교 컴퓨터학과**
{pinesnow, jhm, nerguri, dskim, jwcha, ahnjh, gblee}@nlp.postech.ac.kr
{hskim, kksun}@diquest.com*, seojy@ccs.sogang.ac.kr**

Open-domain Question Answering Using Lexico-Semantic Patterns

Seungwoo Lee, Hanmin Jung, Byung-Kwan Kwak, Dongseok Kim, Jeongwon Cha, Joo
Hui An, Gary Geunbae Lee, Harksoo Kim*, Kyungsun Kim*, Jungyun Seo**
Dept. of Computer Science & Engineering, POSTECH
DiQuest.com*
Dept. of Computer Science, Sogang University**

요 약

본 연구에서는 오픈 도메인에서 동작할 수 있는 질의 응답 시스템(Open-domain Question Answering System)을 구현하고 영어권 TREC에 참가한 결과를 기술하였다. 정답 유형을 18개의 상위 노드를 갖는 계층구조로 분류하였고, 질문 처리에서는 LSP(Lexico-Semantic Pattern)으로 표현된 문법을 사용하여 질문의 정답 유형을 결정하고, lemma 형태와 WordNet 의미, stem 형태의 3가지 유형의 키워드로 구성된 질의를 생성한다. 이 질의를 바탕으로, 패시지 선택에서는 문서검색 엔진에 의해 검색된 문서들을 문장단위로 나눠 점수를 계산하고, 어휘체인(Lexical Chain)을 고려하여 인접한 문장을 결합하여 패시지를 구성하고 순위를 결정한다. 상위 랭크의 패시지를 대상으로, 정답 처리에서는 질문의 정답 유형에 따라 품사와 어휘, 의미 정보로 기술된 LSP 매칭과 AAD (Abbreviation-Appositive-Definition) 처리를 통해 정답을 추출하고 점수를 계산하여 순위를 결정한다.

구현된 시스템의 성능을 평가하기 위해 TREC10 QA Track의 main task의 질문들 중, 200개의 질문에 대해 TREC 방식으로 자체 평가를 한 결과, MRR(Mean Reciprocal Rank)은 0.341로 TREC9의 상위 시스템들과 견줄 만한 성능을 보였다.

1. 서론

인간의 정보에 대한 욕구는 오래 전부터 있어 왔고, 이러한 욕구를 충족시키기 위해 정보의 형태도 데이터베이스를 비롯해, 일반 텍스트, 최근에는 멀티미디어까지 점차 다양해지고 그 양도 많아졌다. 특히 인터넷의 확산으로 사이버 공간 상에 무수히 늘어난 텍스트 자료로부터 정보를 획득하고자 다양한 인터넷 검색엔진들이 등장하였고 또한 사용자로부터 좋은 호응을 얻고 있다. 그러나, 이러한 시스템들은 엄격히 말해서 정보검

색(information retrieval)이 아니라 문서검색(document retrieval)이라고 부르는 것이 옳다[9]. 사용자의 질문에 적합한 내용을 담고 있는 문서를 찾아주는 것이지 찾자 하는 내용 그 자체를 제시해 주는 것은 아니며, 문서 내에서 원하는 내용을 찾는 것은 사용자의 몫이다. 이런 의미에서 정보 검색의 원래 의미에 한 단계 다가서는 기술이 질의-응답 시스템(question answering system)이다.

정보 검색 시스템과 달리, 질의-응답 시스템에서는 단

* 본 연구는 교육부 BK21 과제에 의해 수행된 것임.

순히 키워드를 추출하는 수준이 아니라, 질문의 의미를 파악하여 찾고자 하는 것이 무엇인지를 이해해야 한다. 또한 대용량 문서집합에서 질문에 대한 답을 찾기 위해 각 문서를 분석하여야 한다.

본 연구에서 우리는 먼저 질문에 대한 정답 유형 (Answer Type)[1][2][3][5]을 정의하고, 정답 유형별로 질문의 단어들에 대한 어휘나 WordNet 의미[4], 단어의 순서를 기술한 LSP(Lexico-Semantic Pattern) 문법을 구축하고, 이를 바탕으로 자연어 질의를 분석하여 정답 유형을 결정한다. 또한 질문에서 의미 있는 단어를 골라내어 lemma형태, WordNet 의미, Porter's stem형태 등 3가지 유형의 키워드를 사용하여 문서에서 정답을 담고 있을 패시지(passage)를 선택한다. 이 패시지를 대상으로, 해당하는 정답 유형에 따른 정답 후보들을 매칭하고 이들을 필터링하고 랭킹하여 가장 적합한 정답을 찾는다. 정답 매칭을 위해 각 정답 유형에 따라 품사와 어휘, WordNet 의미 정보로 기술된 LSP를 구축하여 사용한다.

2. 질문 처리 (Question Processing)

질의-응답 시스템의 첫 단계는 사용자의 질문을 해석하는 과정이다. 이 단계에서 가장 중요한 것은 사용자가 찾고자 하는 것이 무엇인지를 파악하는 것이다. 찾고 있는 것이 사람의 이름인지, 지명인지, 단체의 이름인지, 사건이 일어난 때인지, 아니면 그밖에 다른 무엇인지를 파악해야 한다. 다음으로 중요한 것이 찾고자 하는 것에 대한 조건이 무엇인지를 파악하는 것이다. 찾고자 하는 것이 사람의 이름이라면 그 사람이 어떠한 사람인지를 설명하는 핵심이 되는 키워드를 알아 내야 한다.

그림 1은 질문 처리의 각 모듈들을 보여준다. 먼저 태거(tagger; tree tagger[8])가 사용자가 입력한 질문에 품사를 할당하고, Noun Phrase Chunker가 명사구를 탐지한다. Answer Type Determiner는 질문을 LSP(Lexico-Semantic Pattern)와 비교하여 정답 유형(answer type)을 결정하고, Query Formatter는 문서 검색 및 패시지(passage) 선택에서 사용될 키워드를 결정한다.

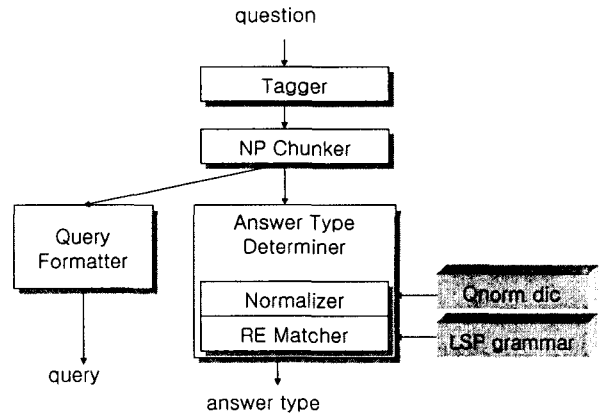


그림 1 질문 처리 (Question Processing)

2.1 정답 유형 (Answer Type)

사실 기반(fact-based)의 질문[9]에 대한 정답이 될 수 있는 유형을 18개의 상위 노드를 갖는 계층구조로 분류하였다. 이 분류는 FALCON[3]에서 제시하고 있는 분류를 참조하여, TREC-8과 TREC-9에서 사용된 질문들과 자체적으로 수집한 질문들을 이용하여 만들었다. "When ... ?" 형태의 질문에 대해서는 년도나 월일, 시간 등의 각각이 정답이 될 수 있으나 "What year ... ?" 라고 했을 때에는 년도만이 정답이 될 수 있기 때문에 정답 유형을 계층 구조로 나누는 것이 유용하다. 다음은 정답 유형의 계층구조에서 상위 18개 노드들이다.

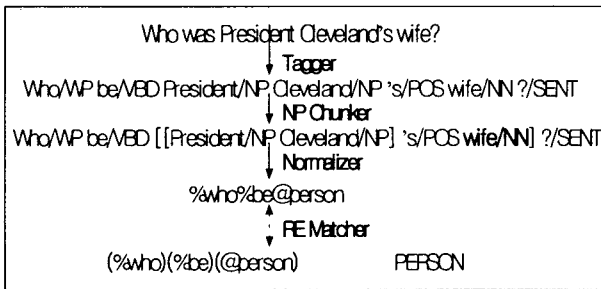
QUANTITY	DATE	TIME	PROPERTY
LANGUAGE_UNIT	LANGUAGE	SYMBOLIC_REP	
ACTION	ACTIVITY	LIFE_FORM	
NATURAL_OBJECT	LOCATION	SUBSTANCE	
ARTIFACT	GROUP	PHENOMENON	STATUS
BODY_PART			

2.2 LSP를 이용한 정답 유형 결정

질문에 대한 정답 유형을 결정하는 데 있어서 의문사가 가장 중요하다. 그러나 질문에서 의문사만으로 정답 유형을 결정할 수 있는 경우는 일부에 지나지 않는다. 대

부분의 경우, 의문사 외에도 의문사 주변의 어휘 또는 그 어휘의 의미들과 이들의 순서들을 함께 파악해야만 그 질문에 대한 정답 유형을 결정할 수 있다. 물론, 의문문 형태가 아닌, 명령문 형태의 질문에서는 의문사 대신 동사와 그 동사의 목적어가 정답 유형 결정에 중요하다.

이러한 특성을 바탕으로 사용자의 질문을 LSP(Lexico-Semantic Pattern)으로 표현하여 정답 유형을 결정할 수 있다. LSP 문법은 LSP로 표현된 조건부와 정답유형을 나타내는 결론부로 구성된다. 예를 들어, 질문 "Who was President Cleveland's wife?"은 "%who%be@person"으로 표현될 수 있다. 여기서 'who'나 'be'는 어휘(lexical)를 나타내고, '@person'은 사람의 하위 개념들을 대표하는 의미를 나타낸다. "President Cleveland's wife"는 품사열 패턴을 이용하는 Noun Phrase Chunker를 통해 head noun인 'wife'가 '@person'으로 변환된다. LSP로 변환된 질문은 LSP rule의 조건부와 정규 표현식(RE)으로 비교하여 정답유형이 결정된다.



TREC-8과 TREC-9의 QA track에서 사용된 질문들과 자체적으로 수집한 질문을 바탕으로 현재 361개의 LSP 문법을 만들었다.

2.3 키워드 추출

키워드는 질문과 유사한 문서를 찾는 IR 시스템의 질의로 사용될 뿐만 아니라, 패시지를 선택할 때에도 사용된다.

- 불용어 제거

질문에 나타나는 단어 중에서 정답을 찾는데 불필요하거나 오히려 방해가 되는 단어들을 제거한다. 568개의 엔트리로 구성된 불용어 리스트와 품사정보를 이용하는 방법 외에 다음의 6가지 휴리스틱을 사용한다.

- of를 포함하는 명사구에서 of 앞의 명사가 그 명사구의 의미적 head가 아닐 때, of 앞의 명사 : What *kind* of dog ... ?
- 's를 포함하는 명사구에서 's다음에 오는 name, nickname : What was the man's *name* who was killed ... ?
- 명령형 질문의 명령 동사 : *Tell* me what city ... ?
- to-부정사를 동반하는 동사 : Where do lobsters *like* to live?
- of 앞의 수사 또는 부사 : Name *one* of the Seven Wonders ... ?, Where does *most* of the marijuana entering ... ?
- 의문사 how 다음의 형용사 또는 부사 : How many people did ... ?

- 키워드 유형

질문에 나타나는 토큰(token) 중에서 불용어를 제외한 나머지가 키워드가 된다. 매칭의 정확도와 키워드 불일치 문제를 고려하여 다음 3가지 유형의 키워드를 사용한다.

- 매칭의 정확도를 위해 토큰의 lemma(base form)를 키워드로 사용한다. 단, smallest나 highest 같은 최상급의 형용사나 부사의 경우에는 토큰을 사용한다
- 키워드 불일치를 줄이기 위해 Porter's stemmer[7]에 의해 생성된 stem을 키워드로 사용한다. 이것은 질문의 inventor와 문서의 invented를 매칭시키기 위해 필요하다. (inventor의 lemma형태는 inventor이지만 stem 형태는 invent이다.)
- 명사구의 WordNet 의미를 키워드로 사용한다. 패시지를 선택할 때, 질문의 ship과 문서의

steamship을 매칭시키기 위해 필요하다.
(steamship은 ship의 하위어다.)

- 키워드 가중치

lemma 키워드는 품사별로 가중치가 결정된다. 고유명사와 대문자로 시작하는 일반명사, 최상급의 형용사나 부사는 높은 가중치를 갖고, 동사와 형용사, 부사는 낮은 가중치를 갖는다. Porter's stemmer에 의해 생성된 키워드는 lemma 키워드 가중치의 일부만큼의 가중치를 갖는다. 명사구의 WordNet 의미[4]는 그 명사구를 구성하는 토큰의 수에 비례하는 가중치를 갖는다.

3. 패시지 선택 (Passage Selection)

IR (Information Retrieval) 시스템이 검색한 문서 내에서 찾고자 하는 정답을 담고 있을 패시지를 선택하여 정답 처리 루틴으로 넘겨 준다. 질문에 포함된 키워드는 문서 내에서 정답이 될 수 있는 후보를 제한하는 역할을 한다. 따라서, 정답은 질의 키워드들과 가까운 거리에 나타날 가능성이 높다고 할 수 있다. 이러한 가정을 바탕으로 검색된 문서를 문장부호(punctuation)에 의해 문장 단위로 분할하고, 매칭되는 키워드의 가중치와 키워드 사이의 거리와 순서를 고려하여 각 문장의 점수를 계산한다. 그런데, 실제 문서에서는 대용어가 자주 사용되기 때문에, 이전 문장에 쓰여진 단어를 참조하는 대용어가 정답인 경우, 한 문장 단위로 정답 추출을 시도했을 때, 실패하게 된다. 이런 경우를 고려하기 위해 어휘 체인(lexical chain)이 존재하는 경우에는 인접한 두 문장을 하나의 패시지로 간주함으로써 패시지 내에서 정답을 찾을 수 있다. 어휘 체인은 참조되는 것과 참조하는 것(대용어) 사이의 연결을 가리킨다. 문장의 점수(Score_{sent})는 두 가지 점수의 합으로 계산된다. Score₁은 매칭된 키워드 가중치의 합이다. 3가지 유형의 질의 키워드를 lemma형태와 WordNet 의미, stem형태의 순으로 비교하여 먼저 매칭되는 것의 가중치를 택한다. Score₂는 매칭된 키워드 사이의 거리와 순서, 매칭된 키워드의 수를 고려한 점수이다. 패시지의 점수는 패시지를 구성하는 문장의 점수를 합한 것으로, 이 점

수에 의해 패시지의 랭킹을 결정한다.

$$Score_{sent} = Score_1 + Score_2$$

$$Score_1 = \sum_i weight(qw_i) \text{ if } qw_i \text{ was appeared in a sentence}$$

$$Score_2 = \frac{\sum_{j=1}^{k-1} \frac{weight(dw_j) + weight(dw_{j+1})}{\alpha \times dist(j, j+1)^2}}{k-1} \times matched_cnt$$

weight(qw_i): weight of query word i

weight(dw_j): weight of query word i,

with which document word j was matched

dist(j, j+1): distance between document j and j+1

matched_cnt: number of query words appeared in a sentence

α: constant

4. 정답 처리

정답 처리는 추출된 패시지 상에서 정답 유형에 맞는 정답 후보들을 매칭하고 이들을 랭킹, 필터링하는 작업을 말한다. 우리는 정답 처리의 성능 향상을 위해서 stemmer[7], 시소러스 (WordNet)[4], 백과 사전용 이용한다.

4.1 시스템 구조

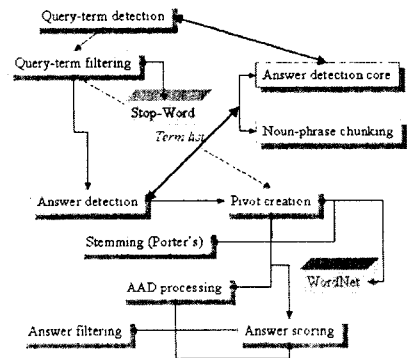


그림 2 정답 처리 시스템 구성도

그림 2는 정답 처리 시스템의 구성을 보여준다. 정답 매칭은 정답 유형을 이용하여 태깅된 패시지 (passage) 내에서 정답 후보들을 발견한다. Pivot은 필터링된 질의 용어 (query term)가 패시지 상에서 나타날 때 생성

하는데, 이것은 패시지 상에서 pivot이 되지 않은 정답 후보들의 랭킹에 이용한다. 정답 유형이 LANGUAGE_UNIT (질의문으로부터 특정한 정답 타입을 결정할 수 없는 경우에 부여되는 정답 타입)인 경우에는 AAD (Abbreviation-Appositive-Definition) 처리를 통해 패시지의 문맥을 파악하고, 정답 후보를 발견한다. 정답 랭킹은 pivot이 아닌 정답 후보들에 대해 다양한 파라미터들을 이용하여 우선 순위를 결정하며, 최종적으로 정답 후보들은 정답의 범위와 유형으로 필터링된다. 이러한 과정은 추출된 각 패시지 별로 반복하며 전체 정답 후보들에 대해 점수에 의한 정렬을 한다.

4.2 정답 매칭

그림 3은 정답 매칭 과정을 보여준다. 정답 매칭을 위해 Lexico-Semantic Pattern (LSP) 문법을 사용하는데, LSP는 품사, 어휘, 그리고 의미 정보로 구성된다. 태깅된 패시지로부터 LSP 문법 매칭을 위한 정규 표현을 구축하기 위해 수사 처리를 포함하는 토큰 그룹화와 태그를 부여하는 미등록어 처리를 전처리로 수행한다.

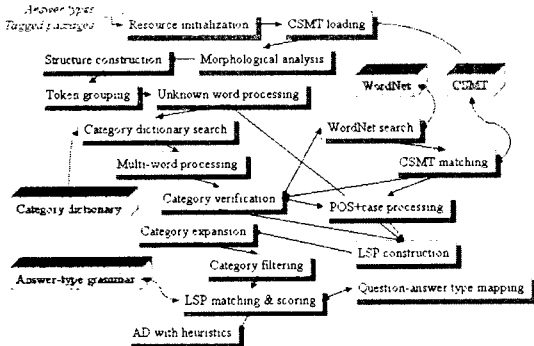


그림 3 정답 매칭 과정

패시지에 대한 정규 표현 구축은 다음의 세 단계로 나누어진다.

- 의미 범주 사전 검색

8만 단어 이상으로 구성된 의미 범주 사전은 하나의 토큰에 대해서 하나 또는 그 이상의 의미 범주를 가진다.

의미 범주는 65개로 나누어지며 정답 매칭을 위한 LSP의 구성 요소가 된다.

- 시소러스 검색

영어 질의를 위해 WordNet[4]을 사용하고 있으며 의미 범주 사전 검색이 실패한 경우에 수행한다. 시소러스 검색을 통해 얻은 의미 코드 값은 Category Sense-code Mapping Table (CSMT)을 이용하여 65개의 의미 범주와 매칭을 시도한다. 매칭 한계값을 넘는 경우에는 해당 의미 범주를 정규 표현 구축을 위해 사용한다.

- POS+case 처리

위의 두 검색을 통해 의미 범주를 얻지 못한 경우에는 품사와 대소문자를 결합하여 정규 표현 구축에 이용한다 (예. "Newton"이 "np"의 품사를 가지는 경우에는 처음 글자가 대문자 이므로 "Np"로 표현). 품사에 대소문자를 이용하는 이유는 영어의 경우에 대소문자가 의미를 가지며 정답 후보의 결정에 대소문자가 중요하게 이용되기 때문이다.

구축된 정규 표현은 정답 매칭 문법 (예. 4-1, 4-2, 4-3)과 매칭을 시도하며 질의 처리에서 넘어온 정답 타입과 비교하여 일치하는 경우에 패시지 내의 문맥을 정답 후보로 결정한다.

cd@unit_lengthcd@unit_length
length|1|4|4 (4-1)

cd@unit_length%per@unit_time
speed|1|4|4 (4-2)

cd@unit_length@direction%of%the@country
location|1|6|6 (4-3)

4.3 Pivot 생성

Pivot은 질의문에서 추출된 질의 용어(query term)가 선택된 패시지 상에서 나타나는 경우에 생성한다. Pivot은 정답 후보가 될 수 없으며 패시지 내의 다른 정답 후보들의 점수를 부여할 때 거리 계산의 파라미터

로 이용한다. 일반적으로 질의 용어가 동일한 형태로 패시지 상에 나타나기도 하지만, 그렇지 않은 경우 (이 형태, 유의어 등)도 많다. 본 시스템에서는 이러한 현상을 고려하여 pivot 생성시에 Porter's stemmer[7]와 시소러스(WordNet)[4]를 이용한다. 또한, 정답 후보가 2단어 이상인 경우에는 부분 매칭을 도입하여 불필요한 정답 후보 생성을 방지한다.

4.4 AAD 처리

Abbreviation-Appositive-Definition (AAD) 처리는 정답 타입이 LANGUAGE_UNIT인 경우에 수행한다. 이러한 처리가 필요한 이유는 질의문으로부터 정답 유형이 무엇인지 알 수 없거나 특정한 정답 유형을 결정할 수 없는 경우가 있기 때문이다. 다음은 정답 유형이 LANGUAGE_UNIT이 되는 질의문의 예이다.

- Why does the moon turn orange?
- What is epilepsy?
- What imaginary line is halfway between the North and South Poles?
- What is done with worn or outdated flags?

이러한 질의문들은 패시지 상에서 pivot, 문장 부호, 구문 관계 등을 이용하여 정답 후보를 결정해야 하며, 그 결과 또한 낮은 성능을 보인다.

우리는 이의 보안을 위해 백과 사전 정보를 이용한다. WordNet의 description으로부터 추출하여 구축한 11만 여 단어의 백과 사전으로부터 질의 용어의 description을 얻고, 불용어를 제거한다. Description 내의 남은 단어들에 대해서는 정답 후보와 의미 거리를 계산하여 정답 후보 순위를 재조정한다.

4.5 정답 랭킹

정답 후보의 기본적인 점수 계산 방법은 Pivot과 정답 후보간의 거리와 LSP로부터 얻은 정답 후보의 기본 점수를 이용한다. 여기에 Pivot과 질의 용어 사이의 매칭

타입/비율, Pivot 사이의 평균 거리, 정답 후보의 의미 범주(특히, AAD 처리를 한 경우)에 따라 점수를 계산한다. 최대 거리를 일정한 값으로 제한하는데 Pivot과의 거리가 어느 값 이상인 정답 후보는 실제 정답일 확률이 적음을 반영한다. 다음은 이를 식으로 표현한 것이다.

$$Score_i = R_{pivot} \cdot \left(1 - \frac{dist_{avg,pivot}}{dist_{max,pivot}}\right) \cdot \frac{S_i}{AADfactor} \cdot \frac{1}{N_p} \cdot \sum_{j=1}^{N_p} r_j \cdot \left(1 - \frac{dist_j}{dist_{max}}\right)$$

R_{pivot} : ratio of matched pivots

$dist_{avg,pivot}$: average distance between pivots

$dist_{max,pivot}$: maximum of distance between pivots

S_i : intermediate score of i th Answer Candidate

$AADfactor$:

if question type is language-unit,

if NE type is AAD, 1

otherwise 4

otherwise 1

$Score_i$: final score of i th Answer Candidate

N_p : number of Pivots

r_j : weight factor of match type of j th Pivot

$dist_j$: distance between j th Pivot and i th Answer Candidate

$dist_{max}$: max value of $dist_j$ (14)

R_{pivot} 은 질의 용어에 대한 Pivot들의 비율로, 예를 들어, 질의 용어로 4개가 있고 이와 완전히 일치하는 Pivot이 3개인 경우 0.75 (3/4)의 값을 가지게 된다. 질의 용어와의 매칭 타입에 따라 이 비율의 값은 조정된다. $Dist_{avg,pivot}$ 은 Pivot들간의 평균 거리로 이 거리가 작으면 작을수록 정답 후보가 좀 더 정답에 가까움을 나타낸다. S_i 는 정답 후보가 LSP에 의해 추출될 때의 점수이다. $AADfactor$ 는 정답 유형이 LANGUAGE_UNIT인 경우에만 정답 후보의 의미 범주가 AAD 처리가 아닌 경우에 대해서 S_i 를 감소시킨다. r_j 는 Pivot의 매칭 타입으로 완전 매칭, stem 매칭, 부분 매칭, 시소러스 매칭에 따라 가중치가 정해진다. $Dist_j$ 는 Pivot과 정답 후보 사

이의 거리로, 둘 사이에 문장 구분이 있을 경우 패널티를 주어 거리를 증가시킨다.

5. 실험

본 시스템의 성능을 평가하기 위해 TREC-10 QA track의 main task에 참가하여 결과를 제출하였고 현재 평가 중에 있다. TREC-10 QA track에서 사용된 문서 집합은 약 3GB의 여러 종류의 신문 기사들로 구성되어 있고, 3가지 task 중, main task에서 사용된 질문은 500개로 구성되어 있다. 500개의 질문을 의문사별로 분류한 것을 표1에 실었다. “What is ... ?” 형태의 개념의 정의를 묻는 질문이 예년에 비해 상당히 많아졌음을 알 수 있다.

표 1 질문 유형 분류

How + 형용사, 부사	32
How do	2
What do	24
What is	244
What/which + 명사	94
When	26
Where	27
Who	47
Why	4
Name a	2

NIST에서 제공하는 상위 1000개의 PRISE 검색엔진[6]의 결과로부터 상위 1000개의 패시지를 선택하고 이것을 대상으로 정답을 추출하였다. NIST에 제출된 정답 후보들은 50바이트 이내로 길이가 제한되고 상위 5개의 후보에 대해 MRR(Mean Reciprocal Rank)값으로 평가된다 [10].

제출된 결과 중에서 질문 1001번에서 1200번까지의 200개 질문에 대해 자체적으로 평가했을 때 MRR값은 0.341이었다. TREC-9와는 달리, 정답이 존재하지 않는 질문도 있는데, 자체 평가에서는 정답의 존재유무를 확인할 수 없기 때문에 “no answer” 라는 답은 틀린 것으로 평가하였다. 이런 점을 감안할 때 TREC-9에 참가한 상

위 시스템들에 견줄 만한 성능이라고 판단할 수 있다.

6. 결론

본 연구에서는 오픈 도메인 질의 응답 시스템을 구현하였다. 정답 유형을 18개의 상위 노드를 갖는 계층구조로 분류하였고, 질문 처리에서는 LSP(Lexico-Semantic Pattern)으로 표현된 규칙을 사용하여 질문의 정답 유형을 결정하고, lemma 형태와 WordNet 의미, stem 형태의 3가지 유형의 키워드로 구성된 질의를 생성하였다. 이 질의를 바탕으로, 패시지 선택에서는 검색 엔진에 의해 검색된 문서들을 문장단위로 점수를 계산하고, 어휘체인(Lexical Chain)을 고려하여 인접한 문장을 결합하여 패시지를 구성하고 순위를 결정하였다. 상위 랭크의 패시지를 대상으로, 정답 처리에서는 질문의 정답 유형에 따라 품사와 어휘, 의미정보로 기술된 LSP 매칭과 AAD (Abbreviation-Appositive-Definition) 처리를 통해 정답을 추출하고 점수를 계산하여 순위를 결정하였다. TREC-10 QA Track의 main task 질문 중에서 200개 질문에 대한 자체 평가 결과, 만족할 만한 결과 (MRR=0.341)를 보였다.

앞으로는 지속적인 지식의 보강과 함께, 사용자의 연속된 질문을 하나의 세션으로 처리할 수 있도록 대응어 처리 기능이 추가되어야 한다. 뿐만 아니라, 널리 쓰이고 있는 검색엔진들처럼 빠른 응답 속도를 보일 수 있는 실용적인 질의-응답 시스템을 위한 연구가 필요하다.

7. 참고 문헌

- [1] Lehnert, W. (1978) The process of question answering: A computer simulation of cognition. Lawrence Erlbaum Associates.
- [2] Graesser, A. C., Lang, K., & Horgan, D. (1988) A taxonomy of question generation. Questioning Exchange, 2, 3-16.
- [3] Harabagiu, S., D. Moldovan, et al. (2000) FALCON: Boosting knowledge for answer engines. In Proceedings of the 9th Text REtrieval Conference (TREC-9).
- [4] Miller, G.A. (1995) WordNet: A lexical database, Communication of the ACM, vol 38: No11, pp 39-41.

- [5] Moldovan, D., S. Harabagiu, et al. (1999) LASSO: A tool for surfing the answer net. In Proceedings of the 8th Text REtrieval Conference (TREC-8).
- [6] NIST PRISE Search Engine:
<http://www.itl.nist.gov/div894/894.02/works/papers/zp2/main.html>
- [7] Porter, M.F. (1980) An algorithm for suffix stripping. Program, 14(3), pp 130-137.
- [8] Schmid, H. (1994) Probabilistic part-of-speech tagging using decision trees, Proceedings of International Conference on New Methods in Language Processing.
- [9] Voorhees, Ellen M. and Dawn M. Tice, (2000) Building a question answering test collection. In Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval
- [10] Voorhees, Ellen M. and Dawn M. Tice, (1999) The TREC-8 question answering track evaluation. In Proceedings of the 8th Text REtrieval Conference (TREC-8).