정보추출을 이용한 질의분석

정한민* 민경구** 성원경* 박동인*
*KISTI 정보시스템부 **서강대학교
*{jhm, wksung, dipark}@kisti.re.kr **kkmin@diquest.com

Query Analysis Using Information Extraction

Hanmin Jung* Kyungkoo Min** Won-Kyung Sung* Dong-In Park*
*Information System Division, KISTI **Sogang University

요 약

본 논문에서는 네비게이션 도메인 상에서의 자연어 질의를 분석하기 위한 방법으로 정보추출을 이용한다. 목적지향성 대화문을 처리하기 위해 도입한 정보추출은 미리 정의된 필드들의 값을 채우는 방식으로 대화를 이끌 수 있도록 한다. Lexico-semantic pattern 기반의 언어처리와 추출/필터링/랭킹 규칙들을 사용하여 강건하면서도 애매성 처리가 용이한 정보추출 기법을 이용한다. 네비게이션 도메인 상에서의 실험은 목적지까지의 이동을 위한 사용자와의 대화집합 256개에 대해 문장레벨 97%의 정확율을 보여준다.

1. 서 론

정보추출은 일반적으로 다양한 형태의 문서 (정형/준정형/비정형)로부터 사용자가 정의한 스키마에 맞게 정보를 추출하여 저장하는 응용분야이다. 웹 문서로부터 상품 정보를 추출하고, 이메일로부터 스케줄 정보를 추출하는 것이 좋은 예이다. WHISK [7]와 POSIE [2]는 자연어 문장까지 처리할 수 있는 능력을 지닌 대표적인 시스템들이다.

WHISK는 슬롯 명과 구분자가 섞인 정규표현 기반의 규칙을 사용한다. 비정형 문서로부터 하나의 규칙으로 다중 슬롯을 추출할 수 있다는 점에서 기존 시스템들보다 우위에 있으나, 시스템이 "* (Nghbr) * (Digit) ' ' Bdrm * '$'"과 같이 강하게 결합된 형태로 슬롯 관계를 표현하기 때문에 슬롯들의 모든 가능한 순서를 학습해야 한다는 데 약점을 가지고 있다. 또한, 자연어 문장을처리하기 위해서는 실용적으로 부담이 되는 구문 분석기를 이용하야 하고, 의미적 지식이 체계적으로 설계되지 않았다는 문제점을 가진다.

POSIE는 위에서 언급한 WHISK의 약점들을 동적 슬롯 그룹화와 lexico-semantic pattern [1] [5] 기반의 언어처리를 통해 극복한다. 다른 정보추출 시스템들에 비해 향상된 성능을 보여주고 있지만, 동적 슬롯 그룹화가 두 개의 슬롯간에 경계를 공유한다는 가정을 하고 있기 때문에, 사용자와의 대화집합이나 이메일과 같이 정보들이 산재되어 나타나는 곳에서는 성능에 한계가 발생한다.

본 논문에서는 정보추출의 적용 분야를 정적인 문서가 아닌 목적지향성의 사용자 질의를 분석하고 대화를 처리하는데도 이용할 수 있다는 것을 보여준다. 이를 위해, 문맥 기반의 세 단계 규칙을 이용하여 질의분석의 성능을 높이고, POSIE의 기본 구조인 LSP 기반 언어처리를 도입하여 강건성을 확보한다. 실험을 위해 네비게이션 도메인 상에서의 사용자와의 대화처리를 위한 질의분석을 수행한다.

본 논문의 구성은 다음과 같다. 2장에서는lexico-semantic pattern을 포함한 관련 지식을 설명하고, 3장에서는 문맥기반 정보추출을, 4장에서는 네비게이션 도메인상에서의 목적지향성 질의분석을 기술한다. 5장에서는 산업자원부 중기거점 과제로서 수행한 텔레매틱스 프로젝트의 일환으로서의 실험 결과를 소개한다.

2. 용 어

2.1 개체명 사전

고유명사 위주로 구성된 개체명 사전의 각 엔트리는 형태소, 구, 또는 의미태그로 구성된다. Lexico-semantic pattern (LSP)은 개체명 사전을 통해 의미 태그들을, 품사 태거를 통해 품사들을 획득한다. 의미태그는 두 가지 타입을 가지는데, 하나는 개념이며, 다른 하나는

그들의 인스턴스이다. "%"로 표현되는 개념은 의미 범주를 의미하며, "@"로 표현되는 인스턴스는 예제 단어를 의미한다.

[테이블 1] 개체명 사전의 예

키워드	의미 태그	비고
도시	%city	개념
서울	@city	인스턴스
조직	%organization	개념
YMCA	@organization	인스턴스

2.2 Lexico-Semantic Pattern

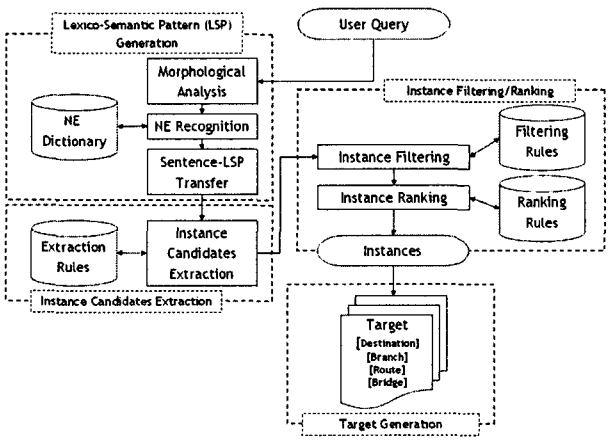
LSP는 하나의 문장에서 형태소 열을 추상화하기 위해 어휘적 엔트리와 의미 타입을 섞어 사용하는 형태를 가진다. 어휘적 엔트리는 형태소, 구, 그리고 품사들을, 의미 타입은 의미 태그와 슬롯들을 포함한다. 문장-LSP 변환은 주어진 문장을 LSP로 변환하는데, 그 예는 다음과 같다.

문장 : 서강대교 건너서 강남구청으로 가자
LSP : @bridge 건너 ef @organization j로 가 ef

LSP는 구/절/문장과 LSP 사이에서의 N : 1 매핑을 통해 추출 적용성을 향상시킨다. LSP 기반의 언어처리 연구가 질의응답, 정보추출 (평생교육/구인/구직/이메일) 등 다양한 도메인에서 수행되었으며, 대표적인 시스템으로 POSIE[2]와 SiteQ [4]가 있다.

3. 정보추출

3.1 시스템 구성



[그림 1] 정보추출 시스템구성도

그림 1은 LSP 기반 언어처리를 포함하는 정보추출 시스템 구성도를 보여준다. LSP 생성 후 인스턴스 후보들은 추출 규칙에 의해 획득되며, 이들은 필터링 규칙에 의해 필터링되고, 랭킹 규칙에 의해 점수가 부여된다. 타깃 생성에서는 네비게이션 질의로부터 추출된 인스턴스들을 네 개의 슬롯에 배치하는 작업을한다. 그림 2는 정보추출을 이용한 질의분석의 예를 보여준다.

[현재 대화 상태]
0000[1] 네 개의 숫자는 목적지, 지점, 경로, 다리에 대응되며, 1인 경우에는 해당 정보가 획득되었음을 의미한다. (초기화)

[입력 질의]
"양화대교는 싫고 서강대교를 건너 강남구청으로 가자"

[LSP 생성 후]
@bridge j 싫 ef @bridge j %pass @organization j로 가 ef

[적용된 추출 규칙]
@bridge ? #bridge
@organization ?#organization

[인스턴스 후보 추출 후]
#bridge (양화대교)
j 싫 ef
#bridge (서강대교)
j %pass
#organization (강남구청)
j로 가 ef

[적용된필터링 규칙]
#bridge j 싫 ? CENTER | #bridge

[인스턴스 필터링 후]
#bridge (양화대교) ? 필터링됨
j 싫 ef
#bridge (서강대교)
j %pass
#organization (강남구청)
j로 가 ef

[적용된랭킹 규칙]
#bridge j %pass ? CENTER | #bridge
j로 가 ef ? LEFT | #organization

1) 네 개의 숫자는 목적지, 지점, 경로, 다리에 대응되며, 1인 경우에는 해당 정보가 획득되었음을 의미한다.

[추출된슬롯들]

슬롯	인스턴트	점수
목적지	강남구청	0.2
지점	없음	
경로	없음	
다리	서강대교	0.2

신규 대화상태 :
100

[그림 2] 질의분석의 예

3.2 LSP 생성

LSP 생성 (그림 1의 우측 상단)은 품사 태거와 개체명 사전을 이용하여 입력 문장을 LSP로 변환한다. 동사와 형용사를 제외한 모든 입력 형태소들은 문장-LSP 변환에 의해 품사들과 의미 태그들로 변환된다.

3.3 인스턴스 후보 추출

인스턴스 후보 추출은 추출 규칙과 이전 단계에서 생성된 LSP와 매칭하는 것을 통해 모든 가능한 인스턴스 후보들을 찾는다. 추출 규칙은 형태소, 품사, 그리고 의미 태그로 구성된 LSP를 왼쪽에, 이에 대응하는 슬롯 명을 오른쪽에 가진다. 이 단계의 결과로서 규칙의 왼쪽과 매칭된 모든 표층형태는 인스턴스 후보로서 해당 슬롯에 할당된다. 그림 2에서는 3개의 인스턴스 후보들 (#bridge(양화대교), #bridge(서강대교), #organization(강남구청))이 추출된다. 추출 규칙은 인스턴스 후보의주변 문맥을 참조하지 않는 반면에, 필터링 규칙과 랭킹 규칙은 추출된 후보들을 검증하기 위해 주변 문맥을 활용한다.

3.4 인스턴스 필터링/랭킹

이전 단계에서 추출된 인스턴스후보들을 줄이기 위해 인스턴스 필터링은 필터링 규칙을 후보들에 적용한다. 인스턴스 랭킹은 인스턴스 필터링에 의해 제거되지 않은 인스턴스 후보들에게 점수를 부여한다. 필터링 규칙과 랭킹 규칙의 왼쪽에는 추출 규칙과 유사한 LSP를 가지지만, 추출 규칙의 오른쪽에 있는 슬롯 명들을 추가적으로 포함하는 확장된 LSP 형태로 구성된다.

3.4.1 필터링 규칙

필터링 규칙의 오른쪽은 연산 범위와 슬롯 명을 가진다. 연산 범위는 CENTER, LEFT, 그리고 RIGHT로 나누어지며, 해당 범위 내의 슬롯 명을 제거하는 데 이용된다. 만일 연산 범위가 LEFT라면, 현재 위치로부터 첫 번째 왼쪽에 나타나는 슬롯 명과 일치하는 인스턴스 후보가 필터링된다. CENTER의 경우에는 현재 위치내의 후보가 필터링된다. 연산 범위는 미리 정의된 윈도우 크기 (현재 8) 내에서만 유효하며, 범위를 벗어나는 경우에는 현재 매칭된 규칙이 효력을 잃는다. 필터링된 후보들은 그들이 항상 유효하지 않은 인스턴스라는 것이 아니라, 현재 문맥 상에서 부적절하다는 의미로 해석될 수 있다. 그림 2에서 "양화대교"와 "성수대교" 모두 다리를 지칭하지만, "#bridge j 싫" 이라는 문맥에서는 "양화대교"만 부적절한 인스턴스로 간주된다.

3.4.2 랭킹 규칙

인스턴스 랭킹은 이전에 추출된 후보들에 점수를 부여함으로써 애매성을 가진 후보들 간에 우선순위를 결정할 수 있도록 한다. 일반적으로 비정형 문서에 효과가 크며, 질의 문에 대해서도 애매성을 가지는 경우가 생기므로 적용되어야 하는 규칙이다. 랭킹 규칙의 오른쪽에는 필터링 규칙과 비교할 때 추가된 연산 정보가 있는데, 이 연산 정보는 슬롯들을 결합하고, 하나의 슬롯을 다른 슬롯으로 대체하고, 여러 슬롯들을 동시에 선택할 수 있도록 한다 (테이블 2)

[테이블 2] 랭킹 규칙에서의 연산 정보들

연산	설명
slot-name1 : slot-name2 :	점수를 주기 위해 연산 범위 내의 여러 슬롯을 선택
TAG>slot-name	왼쪽 품사나 의미 태그를 오른쪽 슬롯 명으로 대체
slot-name1>slot-name2	왼쪽 슬롯 명을 오른쪽 슬롯 명으로 대체
?>slot name	어느 형태소/품사/의미 태그를 오른쪽 슬롯 명으로 대체
#all	점수를 주기 위해 연산 범위내의 모든 슬롯을 선택

다음은 인스턴스 랭킹에서 사용되는 점수 부여 방식이다. 그림 2에서 "j로 가 ef"가 세 개의 LSP 요소로 구성되므로 "강남구청"의 점수는 0.2 (3/15)가 된다.

```
if (랭킹 규칙이 매칭) {
        점수 = LSP 요소의 개수 / 상수[2)]
}
```

4. 질의분석

정보추출은 목적지향성 대화에서의 질의분석에 적합

2) 현재 15

한 방식이다. 하나의 대화 집합 내에서 목적을 달성하기 위해 주요한 정보들을 사용자가 계속 제공을 해나가고, 컴퓨터는 아직 채워지지 않은 정보를 요청하는 방식으로 대화를 이끌어나갈 수 있다. 다음은 네비게이션 도메인에서의 대화 집합 예를 보여준다.

컴퓨터 : 목적지를 말씀해주세요. 사용자 : 서초구청으로 가자. 컴퓨터 : 경로를 선택해 주십시오. 사용자 : 한남대교를 지나서. 컴퓨터 : 네비게이션을 시작합니다.

4.1 네비게이션 정보

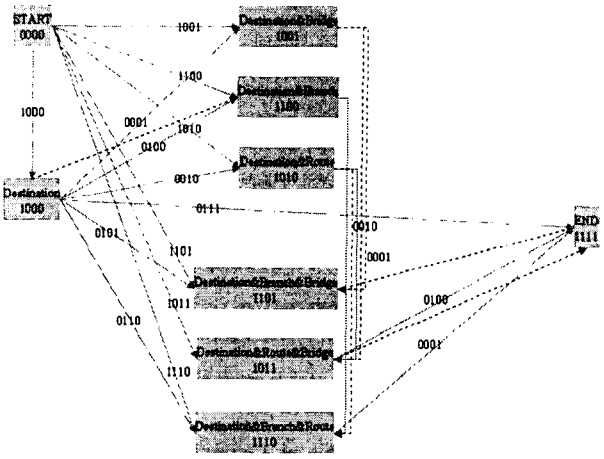
네비게이션 도메인에서 획득해야 할 정보를 본 논문에서는 다음의 네 가지로 정의한다.

목적지 지점 (목적지가 병원인 경우에만 요구됨) 경로 　　최적경로 (다리선택 없음) 　　최단경로 (다리 선택 없음) 　　일반경로 다리 (경로가 일반경로인 경우에만 요구됨)

위 네 가지 정보의 집합을 네 자리 숫자로 표현하여 이를 대화 상태로 정의한다. 만일 네 가지 정보 중 아무것도 획득되지 않은 상태라면, 네 자리 숫자는 "0000"이 된다. 다음은 각 단계에서 가능한 대화 상태를 보여준다.

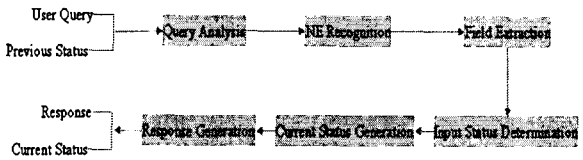
시작 상태 : 0000 종료 상태 : 1111 입력 상태 : 1000, 0100, 0010, 0001, 0101, 0110, 0111, 1100, 1010, 1001, 1110, 1011, 1101 중간 상태 : 1000, 1100, 1010, 1001, 1110, 1011, 1101

이들에 대한 상태변환 다이어그램은 다음과 같다. "1000"에서 "1100"처럼 일부 상태에서는 자동적으로 다음 상태로 이동하기도 하는데, 현재의 획득 정보를 분석하여 특정 정보가 요구되지 않는다면 바로 상태변환을 할 수 있다. 예를 들어, 사용자가 목적지로 병원 이외의 지명을 요구하는경우에는 지점이 필요 없으므로, 지점을 자동으로 세팅하고 상태변환을 한다.



[그림 2] 네비게이션 도메인에서의 상태변환 다이어그램

4.2 정보추출을 이용한 질의분석



[그림 3] 질의분석 흐름도

그림 3은 정보추출을 이용한 질의분석 흐름도를 보여준다. 사용자 질의와 이전 대화 상태가 입력으로 들어오며, 응답 (미 획득 정보 요청 문장)과 업데이트된 상태를 사용자에게 보낸다. 그림 4는 이러한 처리과정을 거쳐 네비게이션을 실행하는 예를 보여준다.

[현재 상태 : 0000] 컴퓨터 : 목적지를 말씀해주세요. 사용자 : 서울대학교병원에 가자. [목적지 : 서울대학교병원] [입력 상태 : 1000] [현재 상태 : 1000] 컴퓨터 : 어느 지점으로 가시고자 합니까? 사용자 : 분당 [지점 : 분당] [입력 상태 : 0100] [현재 상태 : 1100] 컴퓨터 : 경로를 선택해 주십시오. 사용자 : 가능한 빨리.

[경로 : 최적 경로]
[입력 상태 : 0010]
[다리 : 불필요]
[자동상태변환 : 1110 → 1111]

[현재 상태 : 1111]
컴퓨터 : 네비게이션을 시작합니다.

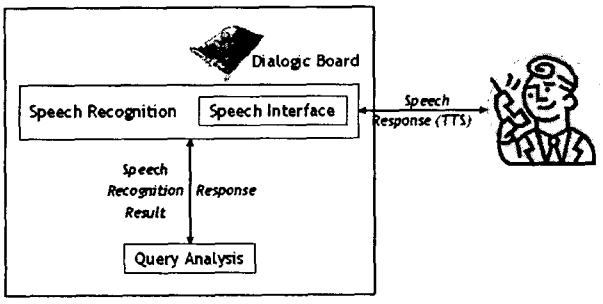
[그림 4] 네비게이션 도메인에서의 질의분석 예

다음은 네비게이션 도메인에서 구축된 추출 규칙의 예를 보여준다.

```
%not 막히 ef %road 로 %most %fast ? #o_method
@school %belong %hospital ? #destination
짧 ef %distance ? #s_method
@city %bridge ? #bridge
@city %company ? #branch
```

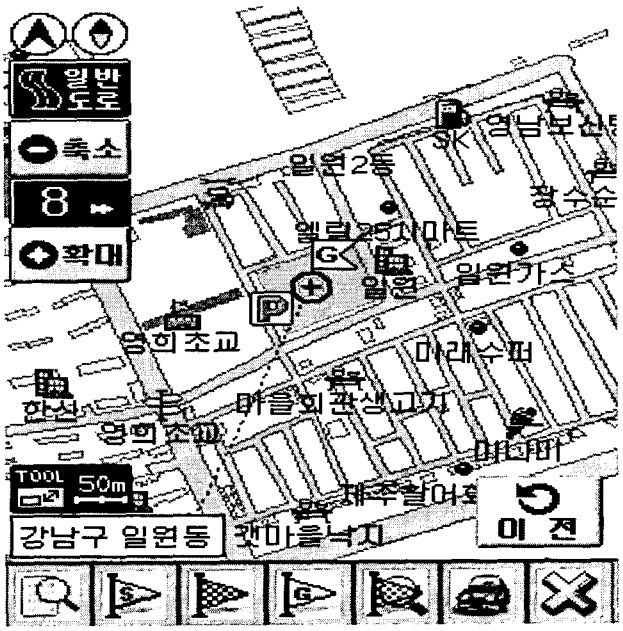
5. 실 험

그림 5는 음성 인터페이스를 가진 실험 환경을 보여준다 [3] [6]. 질의분석은 음성인식된 질의와 이전 대화상태를 입력으로 받고, 응답과 업데이트된 상태를 음성 인터페이스로 돌려주며, 음성 인터페이스는 TTS를 통해 사용자에게 발화한다.



[그림 5] 네비게이션 도메인을 위한 실험 환경

우리는 371개의 사용자 질의문들을 포함하는 256 대화집합을 수작업으로 구축하였으며, 이 집합으로부터53개의 추출 규칙, 1개의 필터링 규칙, 그리고 27개의 랭킹 규칙을 구축하였다. 음성인식 성능은 문장 단위로 75.2%의 성공률을 보여주었으며, 정보추출 과정을 거친 질의분석을 통해 문장 이해도[3]를 97% (음성인식오류수정 후 기준 98%)로 향상시켰다. 실차 환경 176문장에 대해서는 음성인식 성능이56.8%인데 반해, 질의분석기에서의 문장 이해도는 88% (음성인식오류수정 후 기준 97%)를 보여주었다. 이 실험 결과는 음성인식의 오류를 상당한 수준으로 보상할 수 있음을 보여준다. 우리는 실제로 잘못 인식된 기능어, 불필요한 명사 류가 문맥 기반 정보추출에 영향을 미치지 않는다는 것을 알 수 있었다. 그림 6은 PDA 상에서의 네비게이션 예를 보여준다.[4]



[그림 6] 네비게이션 예

6. 결 론

본 논문에서는 정보추출을 단순한 정적 문서에의 적용을 넘어 대화처리를위한 질의분석에 이용함으로써 보다 확장된 영역에서의 가능성을 보여준다. 세 가지 규칙과 LSP 기반 언어처리가 강건성과 정확성이 요구되는 실시간 대화처리에 적합함이 실험을 통해 보여졌다. 앞으로 다양한 분야의 대화처리와 응용분야에 적용함으로써 정보추출의 한계를 극복해 나가고자 한다.

참고 문헌

[1] H. Jung, G. Lee, W. Choi, K. Min, and J. Seo, "Multilingual Question Answering with High Portability on Relational Databases," *IEICE Transactions on Information and Systems*, vol. E86-D, no. 2, 2003.

[2] H. Jung, Rule-based *Information Extraction with Automatic Knowledge Expansion*, Ph.D. Thesis, Pohang University of Science and Technology (POSTECH),

3) 사용자 문장에 대해 정확한 정보를 획득했는가를 의미하는 의미 인식율이다.

4) 본 연구의 일부를 PDA 상에서 테스트한 예임

Korea., 2003.

[3] M. Jung, B. Kim, and G. Lee, "Semantic Oriented Error Correction for Spoken Query Processing," In *Proceedings of IEEE Automatic Speech Recognition and Understanding Workshop*, 2003.

[4] G. Lee, J. Seo, S. Lee, H. Jung, B. Cho, C. Lee, B. Kwak, J. Cha, D. Kim, J. Ahn, H. Kim, and K. Kim, "SiteQ : Engineering High Performance QA System Using Lexico-Semantic Pattern Matching and Shallow NLP," In *Proceedings of the 10th Text Retrieval Conference*, 2001.

[5] A. Mikheev, and S. Finch, "Towards a Workbench for Acquisition of Domain Knowledge from Natural Language," In *Proceedings of the 7th Conference of the European Chapter of the Association for Computational Linguistics*, 1995.

[6] W. Shin and M. Kim, "Feature Vector and Frame Weighting to Improve ASR Robustness in the Noisy Conditions," In *Proceedings of SST-2002*, 2002.

[7] S. Soderland, "Learning information extraction rules for semi-structured and free text," *Machine Learning*, vol. 34, 1999.