

# 관계형 데이터베이스 상에서의 다국어 질의 응답 시스템<sup>(1)</sup>

정한민\*<sup>o</sup> 이근배\* 최원석\*\* 민경구\*\* 서정연\*\*\*  
\*포항공과대학교 컴퓨터공학과  
{jhm, gblee}@postech.ac.kr  
\*\*다이퀘스트닷컴  
{dolhana, kkmin}@diquest.com  
\*\*\*서강대학교 컴퓨터학과  
seojy@ccs.sogang.ac.kr

## A Multi-lingual Question-Answering System on Relational Databases

Hanmin Jung\*<sup>o</sup> Gary Geunbae Lee\* WonSeug Choi\*\* KyungKoo Min\*\* Jungyun Seo\*\*\*  
\*Dept. of Computer Science & Engineering, POSTECH  
\*\*DIQUEST.COM Inc.  
\*\*\*Dept. of Computer Science, Sogang University

### 요 약

본 논문은 자연어 인터페이스에 기반한 관계형 데이터베이스 상에서의 질의 응답 시스템에 대해 기술한다. 본 시스템은 다국어, 다중 도메인, 다중 DBMS를 지원하는 시스템으로, 주로 오디오와 비디오 관련 제품들에 대한 정보를 다룬다. Lexico-semantic pattern (LSP) 문법을 관계형 데이터베이스 상에서의 질의 응답 시스템에 최초로 도입하여 기존의 시스템들에 비해 높은 성능을 보이며, linguistic front-end (LFE)와 database back-end (DBE)를 명확히 구분하고 각종 리소스들을 엔진과 분리함으로써 높은 이식성을 가지도록 한다.

### 1. 서론

관계형 데이터베이스 상에서의 자연어기반 질의 응답 시스템은 사용자가 자연어를 이용하여 DB에 접근할 수 있도록 하는 시스템 (NLIDB라고도 함)으로 [Senturk97], 사용자 질의에 대한 결과로서 자연어, 테이블, 또는 그래프 등을 출력할 수 있다. 다른 DB 인터페이스들로는 특정 DBMS에 맞는 SQL을 이용하는 formal query language 인터페이스와 특정 필드들에 사용자가 패턴을 넣을 수 있도록 하는 폼기반 인터페이스와 키보드와 마우스를 이용하여 테이블에 접근할 수 있도록 하는 그래픽기반 인터페이스 등이 있다.

자연어기반 인터페이스의 장점은 SQL 등의 언어를 배울 필요가 없으며, 부정어나 수량어의 표현이 용이하며 [Cohen91], 담화나 문맥을 지원할 수 있다는 것이다 [Hendrix82].

본 시스템은 이러한 자연어 인터페이스에 기반한 관계형 데이터베이스 상에서의 질의 응답 시스템으로, 한국어와 영어 및 기타 언어로의 확장이 가능한 다국어를 지원하며 다중 도메인과 다중 DBMS 상에서 동작한다. 본 논문은 2장에서의 관련 연구와, 3, 4, 5장에서의 시스템 구조 및 내용, 6장에서의 실험으로 구성된다.

<sup>(1)</sup> 본 연구는 교육부 BK21 프로젝트에 의해 수행된 것임

## 2. 관련 연구

Natural Language Interface to Databases (NLIDB)는 1960년대 말에서 1970년대 초에 걸쳐 나타나기 시작했는데, 그 대표적인 시스템은 월석의 화학적 분석 내용을 담고 있는 DB 상에서의 자연어 인터페이스인 LUNAR [Woods72]이다. 이 당시의 시스템들은 특정 DB를 영두에 두고 만들어진 실험적인 시스템들이었다. 1980년대의 시스템들은 중간 표현 방식을 시도하기 시작하였으며, 이식성에 중점을 두고 다양한 시스템들과의 인터페이스를 고려하였다. 영어 질의를 PROLOG 표현으로 변환한 CHAT-80 [Warren82]과 사용자가 시스템에게 새로운 단어와 개념을 가르칠 수 있도록 한 ASK [Thompson85]가 대표적인 시스템들이다. 1990년대 초에는 GPSG, HPSG, PATR-II와 같은 언어 이론에 기반한 상업적 시스템들이 나타나기 시작하였으며, 도메인 지식의 반자동 구축도 시도되었다. 대표적인 시스템들로는 extraposition 문법과 반자동 도메인 편집기를 사용한 MASQUE/SQL [Androutopoulos93]과 GPSG 문법을 사용한 상용 시스템인 LOQUI [Binot91] 등이 있다. [Demers99]는 NLIDB를 NL-to-SQL 번역 문제로 보고 lexicalist 기계번역 방법론을 사용하였으며, [Meng99]은 UCLA의 CoBase 프로젝트로서 정보 검색 기법과 자연어 인터페이스 기법을 결합한 시도를 하였다. 이 시스템들은 크게 다음과 같이 네 가지 범주로 나눌 수 있다.

패턴 매칭 시스템: 초기 시스템들이 여기에 해당한다. 언어 처리를 배제하여 구현이 용이하지만, 단순한 패턴의 사용으로 인한 성능의 한계를 보여준다 [Shankar00].

구문 기반 시스템: 자연어 질의를 구문 분석하고, 그 결과인 파스 트리를 SQL 질의로 변환하는 문법을 이용한다. 구문 트리와 SQL 질의 간 변환 문법의 구현이 어렵고, 이식성이 떨어진다는 단점이 있다.

의미 문법 시스템: [Waltz78], [Templeton83] 등이 시도한 방식으로 구문 및 의미 처리의 결과로부터 SQL 질의를 생성하는 방식으로 한정된 도메인에서의 빠른 개발에 유용하지만 새로운 도메인에서의 이식성이 떨어진다는 단점이 있다.

중간 표현 언어 방식의 시스템: 최근의 대다수 NLIDB 시스템들이 추구하는 방식으로, 자연어 질의를 로직 형태의 중간 표현 질의로 변환하고 이것을 이용하여 SQL 질의를 생성한다 [Androutopoulos95] [Dale00]. 중간 표현 질의까지를 Linguistic front-end (LFE)라 하며, 그 이후를 database back-end (DBE)라고 한다. LFE는 전처리, 파서, 의미 분석기, 의미 후처리로 구성되며, DBE는 데이터 베이스 언어로의 변환기로 구성된다. 이 시스템 구조는 LFE가 DBMS에 독립적이며 추론 모듈이 LFE와 DBE 사이에 추가될 수 있다는 장점을 가진다. 그렇지만, 의미 후처리기와 같이 파서와 의미 분석기의 성능의 한계로 인한 문제를 보완해야 할 모듈이 필요하며, 그럼에도 불구하고 데이터베이스 응용에서 요구하는 높은 성능을 얻기가 힘들다. 이러한 문제점이 NLIDB 연구가 활성화 되지 못한 주요한 이유들 중의 하나라고 볼 수 있다.

우리는 본 논문을 통해 open-domain 질의 응답 시스템에서 도입하여 높은 성능을 보여주고 있는 lexico-semantic pattern (LSP) [Harabagiu01], [Kim01]을 LFE에서의 언어 분석 및 SQL 질의로의 변환 문법에 이용하는 새로운 구조의 고성능 데이터베이스 기반 질의 응답 시스템을 제안한다. 또한, 본 논문은 동일한 시스템 구조 및 문법 체계를 이용하여 한국어와 영어 질의 응답 시스템을 구축함으로써 언어의 이식성을 보여주며, MySQL과 Oracle의 두 DBMS를 동시에 사용하여 DBMS의 이식성을 보여주며, 서로 다른 스키마의 테이블들을 사용하여 도메인의 이식성을 보여준다.

## 3. 시스템 구조

그림 3.1은 본 시스템의 시스템 구성을 보여준다. 한국어나 영어 질의는 해당 언어를 위한 별개의 형태소 분석을 수행하며, 그 외의 처리는 동일한 모듈을 사용하여 이루어진다.

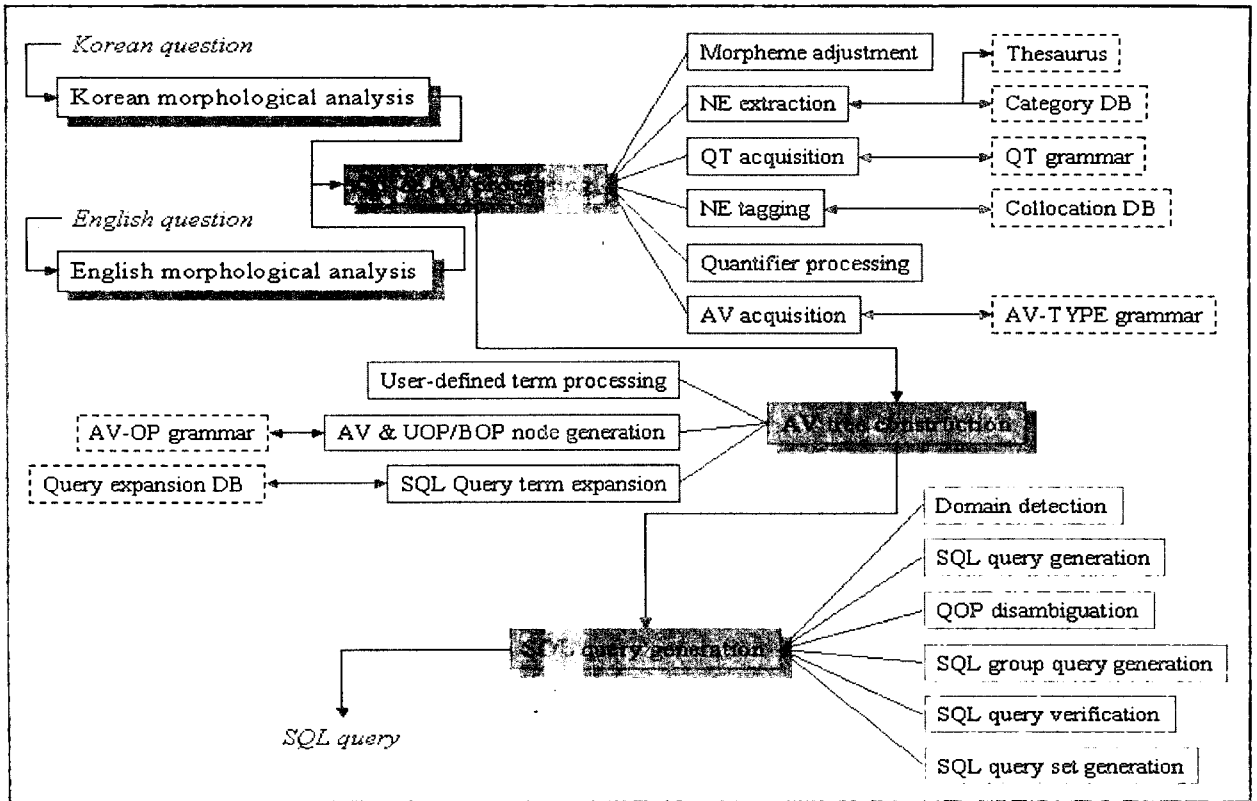


그림 1. 시스템 구조

Linguistic front-end는 질의 처리와 AV (attribute-value) 구조 생성을 포함한다. 질의 처리는 도메인에 맞는 AV 쌍 (필드, 해당 값, 그리고 필드 연산자)들을 생성하는 것이 목적이며, 이를 위해 의미 정보 사전으로부터 named entity (NE)들을 추출하며, 연어 정보를 이용하여 NE 태깅을 수행한다. 또한, 부정어와 수량어를 처리하며, 최종적으로 LSP 패턴으로 구성된 AV-TYPE 문법을 이용하여 AV 쌍들을 얻는다. AV 구조 생성은 SQL 질의 생성을 위한 AV 쌍들과 그들 간의 관계 연산자로 구성된 트리를 만드는 작업을 수행하는데, 사용자 정의 용어 (예. 대형 TV, 고품질 오디오 등)나 개념적 집합을 나타내는 용어 (예. 일본 회사, 국내산 TV 등) 처리를 포함한다.

Database back-end는 SQL 질의 생성을 수행한다. SQL 질의는 도메인과 DBMS에 의존적이며, 데이터베이스로부터 원하는 결과를 가져오기 위한 SQL 문을 생성한다.

본 시스템은 다중 도메인을 사용하므로 사용자가 도메인을 결정하지 않은 경우에는 질의 처리 결과를 이용하여 도메인을 자동으로 결정하며, SQL 문을 "SELECT ... FROM ... WHERE ..." 구조를 기본으로 하여 생성 및 검증한다.

### 3.1 이식성

데이터베이스 상에서의 질의 응답 시스템은 세 가지 측면의 이식성을 고려해서 설계해야 한다.

언어의 이식성: 본 시스템은 형태소 분석기와 리소스들을 제외한 나머지 모듈들에서 언어에 비종속적인 구조를 가진다. 형태소 분석기는 각 언어를 위한 형태소 분석을 위해, 시소러스, 사전, 그리고 문법도 각 언어의 어순, 의미적 차이로 인해 언어별로 마련된다. 그렇지만, LFE의 나머지 부분과 SQL 질의 생성을 위한 DBE는 입

력 언어 (자연어)에 의존하지 않는 언어의 이식성이 높은 구조를 가진다. 새로운 언어가 추가되는 경우에도 역시 언어의 종속성에 따른 모듈의 구분이 확실하다는 장점을 가지는 구조이다.

**DBMS의 이식성:** DBMS가 바뀌게 되면 DBMS에 의존적인 SQL이 다르게 되는데, 이 SQL을 생성하는 DBE가 영향을 받는다. SQL 질의 생성에서는 각 DBMS에 맞는 SQL을 생성하는데 이는 LFE와는 독립적이다. 본 시스템은 새로운 DBMS의 추가에도 불구하고 SQL 질의 생성에서 필요한 세부 모듈들만 영향을 받는, 높은 DBMS의 이식성을 갖는다.

**도메인의 이식성:** 새로운 스키마를 가지는 도메인에 적용하고자 하는 경우에는 LFE의 리소스들과 DBE가 영향을 받는다. 리소스들 중에서는 해당 도메인에 맞는 의미 정보를 가진 도메인 의미 사전과 새로운 스키마에 포함되는 필드를 포함하는 문법이 추가되어야 하며, DBE에는 새로운 테이블명과 필드명을 요구하는 SQL 질의 생성이 추가되어야 한다. 그렇지만, 본 시스템은 도메인의 변경이 LFE의 질의 처리와 AV 구조 생성에는 영향을 미치지 않는 도메인의 이식성이 높은 구조를 가진다.

#### 4. Linguistic Front-End

Linguistic front-end는 사용자로부터 입력 질의를 받아 형태소 분석한 후에, SQL 질의 생성을 위한 필드-값의 쌍과 필드 연산자, 그리고 그 쌍들을 연결하는 관계 연산자로 구성된 AV 구조를 생성한다.

##### 4.1 LSP 문법

Lexico-semantic pattern (LSP)은 기존의 어휘, 유의어, 품사 정보 등으로 구성된 lexico-syntactic pattern [Kim01]에 비해 의미 정보와 시소러스의 의미 코드가 추가되어 개념적 표현이 가능하며, 정보의 세밀한 기술이 가능하다는 장점을 가진다. 다음은 정규 표현 형식을 가진 LSP 문법의 예를 보여준다 (@ 부분은 의미 정보, 괄호 부분은 WordNet의 의미 코드).

```
%urlwdt%sell@type%and@type%together
```

```
qo_intersectqt_url (1-1)
```

```
%corp%make@type%and@type%together
```

```
qo_intersectqt_corp (1-2)
```

```
^@corpjca(010A6M0R030N0E)xspes$
```

```
@corp|vo_like|5 (1-3)
```

의미 정보는 키가 단어가 되고, 내용이 의미 정보가 되는 의미 범주 사전을 통해 얻으며, 의미 코드는 시소러스 (한국어는 K-WordNet [Lee00], 영어는 WordNet)를 통해 얻는다. 도메인에 따른 이식성을 높이기 위해 의미 범주 사전은 일반 의미 범주 사전과 각 데이터베이스에 맞는 도메인 의미 범주 사전으로 나눈다.

##### 4.2 질의 처리

질의 처리는 형태소 분석된 결과로부터 AV 구조 생성을 위한 필드-값의 쌍과 필드 연산자들을 추출하는 작업을 한다. 이 작업은 named entity 추출 및 태깅, 질의 타입 결정, 부정어/수량어 처리, 필드-값의 애매성 해소를 포함한다.

Named entity (NE)는 의미 범주 사전을 이용하여 얻는데, 의미 범주는 대상 도메인이 가지는 필드와 AV 타입 문법에서 사용하는 의미들로 구성된다 (도메인 사전의 경우). 한 단어가 여러 NE들을 가지는 경우에는 예제 질의들로부터 생성된 트라이그램 형태의 연어 정보를 이용하여 NE 태깅을 수행한다. 입력 질의에서의 단어는 여러 이형태 (영어의 경우에는 대소문자 혼용도 포함됨)로 나타날 수 있는데 (예. 텔레비전, 테레비전, 테레비, 텔레비 등), 이들에 대해서는 의미 범주 사전을 이용하여 정규화시킨다. 질의 타입은 LSP 문법 형식을 가지는 질의 타입 문법 (1-1, 1-2 참조)을 통해 결정되며, 질의 연산자 (MIN, MAX, SUM, AVG 등의 10 개 연산자들로 구성)와 테이블 내 필드의 조합으로 이루어진다. 이 질의 타입은 사용자가 원하는 필드와 결과 형태에 맞추어 결과를 출력하기 위해 사용한다. 필드-값의 쌍 및 필드

연산자는 AV 타입 문법 (1-3 참조)을 이용하여 결정한다. LSP 형식으로 바뀐 입력 질의는 AV 타입 규칙과 매칭을 시도한다. 매칭에 성공하면 필드-값의 쌍과 필드 연산자 (LOWER, UPPER, BETWEEN, LIKE, RANGE, GROUP 등의 14개 연산자들로 구성)를 얻을 수 있다.

### 4.3 AV 구조 생성

AV 구조 생성은 필드-값 쌍들을 이용하여 이들간의 계층 구조를 정하고 이를 트리 형태로 생성하는 작업을 수행한다. 사용자 정의 용어나 개념적 집합 용어는 이 단계에서 구체화 또는 확장되어 AV 구조에 포함된다. 필드-값 쌍들로부터의 구조 생성을 위해 AV-OP 문법을 사용하는데 그 형식은 “필드열 + 스택 구조의 AND/OR 열”이다. 질의 처리에서 얻은 부정어와 수량어는 자신의 범위에 포함되는 AV 하위 구조를 포함하는 형식으로 표현된다. 예를 들어, 부정어의 범위가 2번째 단어부터 5번째 단어였다면, 2번째 단어부터 5번째 단어가거나 이에 포함되는 부속 트리를 자신의 하위 구조로 갖는다. 또한, 테이블 내에서 다양한 이형태를 가지는 단어들에 대해서는 사전에 기반한 질의 확장을 수행한다.

## 5. Database Back-End

도메인과 DBMS의 영향을 받는 Database back-end는 언어 처리 후의 결과인 AV 구조를 이용하여 DBMS를 호출할 SQL 질의를 생성한다.

### 5.1 SQL 질의 생성

SQL 질의 생성은 현재 사용중인 도메인과 DBMS에 의해 영향을 받는다. 다중 도메인인 경우에는 사용자가 특정 도메인을 선택하지 않았다면 시스템이 자동적으로 사용자가 요구하는 도메인을 결정하여 그에 맞는 SQL 질의를 생성하는 것이 바람직하다. 본 시스템에서는 AV 구조에 속하는 각 필드들이 어느 도메인에 속하느냐를 먼저 결정하고 이들 도메인들의 조합을 이용하여 최종 도메인을 결정한다. 만일 결정된 도메인이 양쪽 모두인 경우에는 사용자가 JOIN을 결정한 경우를 제외하고는 JOIN을 하지 않고 결과를 분리하여 보여준다. 질의 타

입이 질의 처리에서 결정되지 못한 경우에는 도메인 결정을 통해 얻은 도메인 정보를 이용하여 올바른 질의 타입을 결정한다.

재질의 경우에는 이전 질의를 현재 생성 중인 SQL의 적절한 위치에 삽입하고, 그렇지 않은 경우에는 질의 타입 처리 (SELECT와 FROM 사이), 테이블 처리 (FROM과 WHERE 사이), 제약 조건 처리 (WHERE 이하) 순으로 SQL 질의를 생성한다. 다중 질의인 경우에는 SQL 질의 간 결합을 시도한다. 제약 조건 처리에서는 스택 구조를 가지는 AV 구조를 이용하여 제약 조건을 생성한다. 최종적으로 SQL 형태를 검증하고 DBMS를 호출한다.

## 6. 구현 및 실험

본 논문에서 기술한 구현 시스템은 웹 인터페이스를 가진 관계형 데이터베이스 기반의 자연어 질의 응답 시스템이다. 대상 언어는 한국어와 영어이며, 도메인은 주로 오디오와 비디오 관련 제품들의 정보로 구성된다. 데이터베이스 정보는 SAA 시스템 [Kim02]을 이용하여 자동 구축한 418 엔트리의 테이블과 가격 비교 사이트인 BestBuyer (<http://www.bestbuyer.co.kr/mainbbr/index.php3>)에서 추출한 1964 엔트리의 테이블이다. 두 DB는 서로 다른 스키마를 가진다 (부록 2). DBMS는 Oracle 8.0.5와 MySQL 3.23.22를 사용하였다. 그림 2는 웹과 본 시스템 간의 연동을 보여주는데, 웹 브라우저를 통한 사용자 질의는 PHP로부터 호출되는 질의 응답 시스템에서 처리되고, 그 결과인 SQL 질의는 CGI를 거쳐 DBMS로 보내진다. 데이터베이스로부터 얻어진 결과는 사용자에게 웹 브라우저를 통해 보여지며, 사용자는 새로운 질의나 결과를 이용한 재질의 (문맥 기반 연속 질의)를 할 수 있다.

실험에 사용한 질의는 한국어와 영어에 대해 5명의 대학원생이 작성한 각 192 문장이다. 이 문장들은 부정어, 수량어, 복합 AND/OR, 질의 결합, 대소문자 혼용, 사용자 정의 용어, 다양한 연산 (예. 값의 범위 및 크기, 문

자연 매칭 방법 등), 이형태 등을 포함한다. 이번 실험에서는 테스트 질의 집합을 별도로 정하지 않아 192 예제 문장들에 대해서만 튜닝을 한 상태이다. 현재, 로그를 통해 10만 여 질의 문장들에서 필터링된 1385 질의 문장을 확보하였으며 이를 테스트 질의로 사용할 예정이다. 부록 3은 한국어와 영어에 대한 192 예제 문장들 중의 일부를, 부록 4는 실험 예를 보여준다.

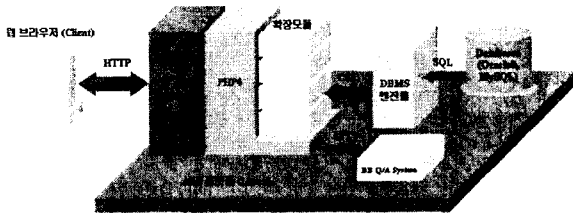


그림 2. 웹과 질의 응답 시스템 간의 연동도

## 7. 결론

본 시스템은 관계형 데이터베이스 상에서의 자연어 질의 응답 시스템으로, LSP 문법의 첫 적용을 통한 성능 향상과 다국어, 다중 도메인, 다중 DBMS를 고려한 시스템 구조를 통한 높은 이식성을 가진다. 또한, 필드 조합을 이용한 자동 도메인 결정, 연어 정보를 이용한 필드 매개성 해소, N-ary 연산자로부터의 정밀한 AV 구조 생성, 부정어 및 수량어 처리, 사용자 정의 용어 및 개념적 집합 용어 처리, SQL 질의 결합 등의 다양한 기능들을 포함한다.

앞으로의 연구는 국어와 영어 이외의 일본어, 중국어에 대한 언어 확장과 관계형 데이터베이스 내용으로부터의 자동 의미 정보 추출에 초점을 맞출 예정이다.

## 참고 문헌

[Androustopoulos93] I. Androustopoulos, G. Ritchie, and P. Thanisch, An Efficient and Portable Natural Language Query Interface for Relational Databases, *Proceedings of the 6<sup>th</sup> International Conference on Industrial & Engineering Applications of Artificial Intelligence and Expert Systems*, 1993.  
 [Androustopoulos95] I. Androustopoulos, G. Ritchie, and P.

Thanisch, Natural Language Interfaces to Databases – An Introduction, *Natural Language Engineering*, 1:1, 1995.

[Binot91] J. Binot, L. Debillé, D. Sedlock, and B. Vandecapelle, Natural Language Interfaces: A New Philosophy, *SunExpert Magazine*, 1991.

[Cohen91] P. Cohen, The Role of Natural Language in a Multimodal Interface, Technical Note 514, Computer Dialogue Laboratory, SRI International, 1991.

[Dale00] R. Dale, H. Moisl, and H. Somers (Eds.), *A Handbook of Natural Language Processing*, Marcel Dekker Inc., 2000.

[Demers99] P. Demers, A Lexical Approach to Natural Language Front-end Database, <http://www.cs.stu.ca/research/groups/NLL>, 1999.

[Harabagiu01] S. Harabagiu, D. Moldovan, M. Pasca, R. Mihalcea, M. Surdeanu, R. Bunescu, R. Girju, V. Rus, and P. Morarescu, The Role of Lexico-Semantic Feedback in Open-Domain Textual Question-Answering, *Proceedings of the 39<sup>th</sup> Annual Meeting and 10<sup>th</sup> Conference of the European Chapter*, 2001.

[Hendrix82] G. Hendrix, Natural Language Interface (Panel), *Computational Linguistics*, 8:2, 1982.

[Kim01] H. Kim, K. Kim, G. Lee, and J. Seo, MAYA: A Fast Question-answering System Based on a Predictive Answer Indexer, *Proceedings of the Workshop Open-Domain Question Answering, Proceedings of the 39<sup>th</sup> Annual Meeting of ACL*, 2001.

[Kim02] D. Kim, J. Cha, G. Lee, Learning mDTD Extraction Patterns for Semi-structured Web Information Extraction, *Will be appeared in IJCPOL*, 2002.

[Lee00] C. Lee, G. Lee, J. Seo, Automatic WordNet Mapping using Word Sense Disambiguation, *Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, 2000.

[Meng99] F. Meng and W. Chu, Database Query Formation from Natural Language using Semantic Modeling and Statistical Keyword Meaning Disambiguation, CSD-TR 990003, University of California, 1999.

[Senturk97] C. Senturk, Natural Language Interface to Databases, In the course of Digital Libraries, E6998-003, 1997.

[Shankar00] A. Shankar and W. Yung, gNarLI: A practical Approach to Natural Language Interfaces to Databases, Term Report, Harvard University, 2000.

[Templeton83] M. Templeton and J. Burger, Problems in Natural Language Interface to DBMS with Examples from EUFID, *Proceedings of the 1st Conference on Applied Natural Language Processing*, 1983.

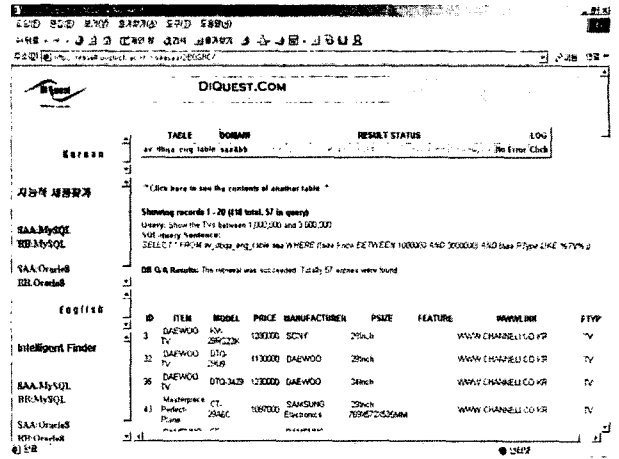
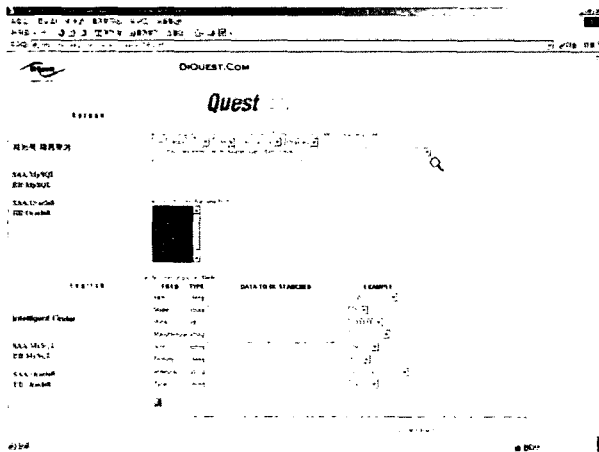
[Thompson85] B. Thompson and F. Thompson, ASK is Transportable in Half a Dozen Ways, *ACM Transactions on Office Information Systems*, 3:2, 1985.

[Waltz78] D. Waltz, An English Language Question Answering System for a Large Relational Database, *Communications of the ACM*, 21:7, 1978.

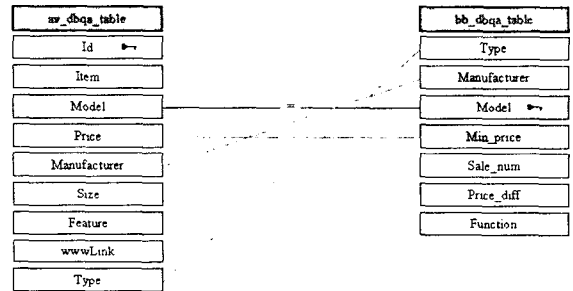
[Warren82] D. Warren and F. Pereira, An Efficient Easily Adaptable System for Interpreting Natural Language Queries, *Computational Linguistics*, 8:3-4, 1982.

[Woods72] W. Woods, R. Kaplan, and B. Webber, The Lunar Sciences Natural Language Information System: Final Report, BBN Report 2378, Bolt Beranek and Newman Inc., Cambridge, Massachusetts, 1972.

## 부록 1. 구현 시스템



## 부록 2. DB 스키마



## 부록 3. 예제 질의

[한국어]

비디오 티비는 가격이 어느 정도 해요?

'신랑각시'라는 모델이 있던데.

100만원 이상의 제품들을 보고 싶는데요

10인치짜리 텔레비전도 있나요?

200만원 이하 프로젝션 텔레비전.

20인치부터 25인치까지의 TV들을 보여주세요

25인치, 21인치, 29인치 제품 중에서 삼성 것만을 골라서 보여줘.

25인치에서 30인치 사이의 TV는 어떤 것이 있나요?

29인치 그리고 슈퍼블랙브라운관 그리고 음성다중 그리고 타이머 기능.

29인치이상의 TV를 파는 판매점은?

30만원 내외에서 TV를 사고 싶습니다. 좋은 제품의 추천을 부탁드립니다.

30인치 TV는 얼마면 살 수 있나요?

30인치 내외의 것은 뭐가 있죠?

33인치이상의 TV들을 모두 보고 싶습니다

34인치 제품 중에서 가장 싼 것은 얼마예요?

4.5백만원 대의 고화질 TV 제품은?

70만원 안밖의 TV들을 보고 싶은데 가능한가요

C29-F100의 가격 및 판매처는?

CK-2520N을 파는 회사들 중 제일 싼 곳은?

CN 시리즈 중에서 가장 큰 것은 몇인치 인가요?

[영어]

A television with caption function among Samsung products.

Show three hundred thousand being-far-reaching TVs that was made in Samsung.

29 inches TV made in Samsung?

Show videos that Samsung does not make.

What is the cheapest model among televisions that Samsung produces?

Find all televisions that are not kinds of Samsung or LG electronics products.

Show products that are not Samsung Electronics TV.

I want to see TV list except the Samsung Electronics.

Show whole TVs that were made in Samsung Electronics.

Are there smaller things than 29 inches among Samsung products?

Find Sony TV.

Find Japan television of imported things.

Find from things that head of a family price is cheap among imported products.

Find audio and video that is not imported.

Product that maximum output is over 400W?

What is the most popular TV among new products? Show Samsung product if possible.

Show all TV products for a wedded pair.

#### 부록 4. 실행 예

입력 문장: "25인치, 21인치, 29인치 제품 중에서 삼성 것만을 골라서 보여줘."

형태소 분석 및 NE 태깅 후의 정규 표현:

```
num@길이단위spnum@길이단위spnum@길이단위
%productnbn%from@corp%것jpvepef@요구pvepfsf
```

질의 처리 결과:

QUERY INFO

Query Number: 1

YesNo: No

query-idx 0

QT : UNKNOWN

qop = NULL

AV 구조 생성 결과:

[0] [0->1] AV size 25인치 like

[1] [3->4] AV size 21인치 like

[2] [0->4] BO or

[3] [6->7] AV size 29인치 like

[4] [0->7] BO or

[5] [11->11] AV corp 삼성전자 equal

[6] [0->11] BO and

SQL 질의 생성 결과:

```
SELECT count(*) FROM (SELECT * FROM
av_dbqa_table saa WHERE (((saa.PSize LIKE '%21인치%') OR (saa.PSize LIKE '%25인치%')) OR (saa.PSize LIKE '%29인치%')) AND (saa.Manufacturer='삼성전자'))
```