

인적사항 데이터베이스 조회를 위한 퍼지질의 시스템의 FSQL 처리

신세영, 정은영, 김승권, 김수영, 박순철
전북대학교 정보통신공학과
e-mail: gracia@internet.chonbuk.ac.kr

FSQL Processing of Fuzzy Query System for a Personal Information Database

Se Young Shin, Eun Young Cheong, Seung Gwon Kim,
Su Young Kim, Soon C. Park
Dept. of Infomation and Communication Eng.,
Chonbuk National University

요약

일반 사용자들의 데이터베이스 사용 증가와 부정확한 질의를 처리할 수 있는 데이터베이스의 요구가 증가하면서 기존의 관계형 데이터베이스에 퍼지이론을 적용한 퍼지 데이터베이스에 대한 연구가 진행되고 있다. 본 논문에서는 인적사항 관련 데이터베이스에 대하여 이러한 퍼지 데이터베이스인 퍼지질의 시스템의 개발을 위해서 아직 일반화 되지 않은 FSQL을 정의하였으며 meta DB의 분류와 그에 따른 소속함수에 대하여 정의하였다. 또한 정의된 데이터를 이용하여 내부에서 필요로 하는 FSQL 처리 부분을 설계하였다.

1. 서론

데이터베이스 시스템은 대용량의 데이터를 저장하고 관리하는 시스템으로 컴퓨터의 가장 성공적인 응용분야 중의 하나이다. 데이터베이스 시스템은 컴퓨터 산업의 발달과 더불어 지속적인 발전을 하였으며, 더욱이 최근의 멀티미디어의 발달과 인터넷 사용의 확장은 데이터베이스 사용을 급증하게 하였다. 이러한 데이터베이스는 Codd가 제안한 관계형 데이터베이스 시스템을 근간으로 하고 있으며 관계형 데이터베이스 시스템을 유지 관리 및 접근하기 위해서 필수적인 언어는 SQL(Structured Query Language)이다. 이러한 데이터베이스 시스템은 비숙련된 사용자라 할지라도 데이터베이스를 접근하기 위해서는 SQL에 익숙해 있어야 하며 사용자가 정확한 질의 조건을 생성시키지 못한다면 해당 데이터베이스로부터 어떠한 정보도 얻지 못한다. 이러한 이유로 최근 국내외에서는 쉬운 질의와 지능적인 결과를 제공하기 위해서 1965년 캘리포니아 대학의 자데(zadeh) 교수 [3,5] 에 의해 처음 도입되어 계속

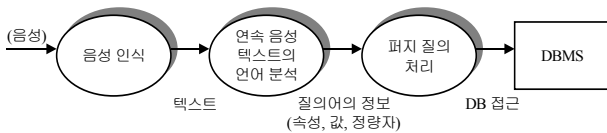
발전되고 있는 퍼지이론을 적용하여 부정확한 질의를 지원할 수 있는 FSQL(SQL)에 대한 연구가 진행중에 있다. [4,6,7,9,10,11,12,13]

본 논문은 인적 사항 데이터베이스를 위한 음성 인터페이스 퍼지 질의처리 시스템을 구현하는 것을 최종목표로 하는 전체과제 가운데 FSQL 처리부분에 대한 연구를 중심으로 작성되었다. 그러나 현재는 방대하고 다양한 모든 데이터베이스를 처리하기에는 퍼지질의의 정형화와 범용화가 불가능하다. 본 논문에서는 웹 상에서 유용한 정보가 될 수 있는 인적사항 항목에 대한 퍼지질의의 처리기를 구축하고자 한다. 본 연구를 통하여 첫 번째로 아직 일반화 되지 않아 기존의 데이터베이스와 호환성이 부족했던 FSQL에 대하여 FSQL을 정의하였다. 두 번째로 지금까지 문제가 되었던 방대한 양의 metaDB(MetaDatabase)로 인한 성능의 저하, metaDB를 구축하기 위한 데이터들 사이의 유사도를 체계적으로 명세하는 방법에 관한 연구 미약함의 문제점을 해결하기 위하여 metaDB의 분류와 그에 따른 소속함수에 대하여 정의하였다. 세 번째로, 정의된

데이터를 이용하여 내부에서 필요로 하는 간단한 FSQL 처리 부분을 설계 및 구현하였다.

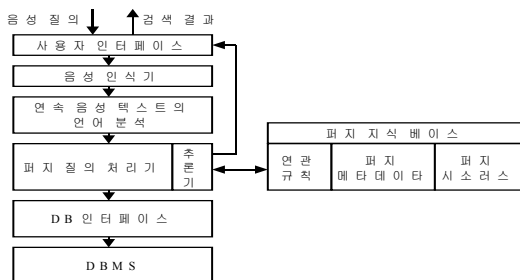
2. 전체 시스템의 구조

전체 시스템의 목표는 인적 사항 데이터베이스를 위한 음성 인터페이스 퍼지질의 시스템을 구축하는 데에 있다. 전체 시스템에서 실질적으로 FSQL 처리를 하려면 아래 그림과 같이 음성인식 단계와 자연어 처리를 통한 연속음성 텍스트의 언어 분석 단계를 거쳐야 한다. 예를 들어 음성을 통하여 “봉급이 많은 사람을 찾아라” [1,2,4]라는 질의가 입력되면, 음성인식 분야에서 음성을 인식하여 텍스트로 변환하고, 연속음성 텍스트 언어분석 단계에서는 Parser와 자연어 처리를 통하여 FSQL 처리부분에서 사용될 심볼 테이블의 키워드 즉, “봉급”, “많은”의 단어를 찾게 된다. 또한 키워드를 검색하여 퍼지 성분이 있으면 FSQL 처리를, 없으면 일반 SQL 처리를 하면 된다. 앞의 예제에서는 “많은”이라는 퍼지 성분이 포함되어 있으므로 FSQL 처리기에서 “봉급”에 관련된 metaDB를 통하여 “많은”의 정도를 나타내주는 데이터를 얻을 수 있다.



[그림 1] 인적사항 데이터베이스를 위한 음성 인터페이스 퍼지질의 시스템의 개요

[그림 1]에서 본 전체 시스템의 단계별 과정은 [그림 2]와 같다. 먼저 input으로는 모든 컴퓨터 시스템에 대하여 가장 이상적인 사용자 인터페이스인 음성을 통한 자연어 질의가 들어온다. 데이터베이스에 대한 질의문으로 자연어 문장이 사용됨으로써, 질의 언어 문법에 대한 전문적 지식을 갖지 않은 일반 사용자들도 DBMS에 대한 용이한 접근이 가능하다. 이를 위해 음성 신호를 일단 텍스트로 변환한 후에 이 텍스트에 대하여 자연어 처리를 하여야 한다. 아울러 연속음성 텍스트는 단순히 신호처리를 거쳐서 나온 결과이기 때문에 언어처리기술을 이용하여 정련을 해 주어야 한다. 또한, 테스트 처리를 통한 결과를 데이터베이스가 인식할 수 있는 속성값, 정량자의 트리플로 변환시켜 주어야 한다. 마지막으로 이러한 트리플로 퍼지 질의를 위한 질의문을 구성하고 퍼지한 의미처리뿐 아니라 시소러스를 통한 질의 확장과 데이터 연관 규칙을 통한 질의처리를 수행하여 DB 인터페이스를 통해 사용자의 질의에 대한 검색 결과를 사용자 인터페이스를 통해 제공한다.



[그림 2] 인적사항 데이터베이스를 위한 음성 인터페이스 퍼지질의 시스템의 구조

본 논문에서는 전체 시스템 중 FSQL을 정의하고 meta DB의 분류와 그에 따른 소속함수에 대하여 정의한다. 또한 간단한 FSQL 처리기를 구현하여 연속음성 텍스트의 언어분석 처리와 현재 나와 있는 음성인식 기술과의 접목 가능성을 보이고자 한다.

3. FSQL 처리

3.1. FSQL

3.1.1. Fuzzy 관계대수

관계대수의 집합-연산 대수는 기본연산과 이를 확장-응용한 확장연산으로 나눌 수 있다. fundamental operation으로는 UNION, DIFFERENCE, CAERTESIAN-PRODUCT가 있으며, Additional operation으로는 INTERSECTION, NATURAL-JOIN 이 있다. 본 논문에서는 Zadeh 교수가 제안한 fuzzy 연산을 따른다. [1,2,3,4]

① UNION

$$\mu_{R \cup S}(t) = \max(\mu_R(t), \mu_S(t))$$

② DIFFERENCE

$$\mu_{R-S}(t) = \min(\mu_R(t), (1 - \mu_S(t)))$$

③ CARTESIAN-PRODUCT

$$\mu_{R_1 \times R_2 \times \dots \times R_n}(t) = \min(\mu_{R_1}(t_1), \mu_{R_2}(t_2), \dots, \mu_{R_n}(t_n))$$

$$\text{where } t = (t_1, t_2, \dots, t_n)$$

④ INTERSECTION

INTERSECTION 은 DIFFERENCE의 확장 연산자이다. 왜냐하면 INTERSECTION 연산자는 DIFFERENCE 연산자로 표현이 가능하기 때문이다.

$$R \cap S = R - (R - S)$$

위와 같은 원리에 의해 다음과 같다.

$$\mu_{R \cap S}(t) = \min(\mu_R(t), \mu_S(t))$$

⑤ NATURAL-JOIN

NATURAL-JOIN 역시 CARTESIAN-PRODUCT의 확장 연산자이므로 관계대수가 동일하다. 차이점은 NATURAL-JOIN의 경우 일치하고자 원하는 attribute에 대한 조건문이 들어가게 되는데, 본 논문의 시스템은 RDBMS를 사용하기로 하였으므로 일치를 원하는 attribute에 대한 퍼지성질을 허용하지 않는다. 즉, 퍼지질의를 나타내는 다음과 같은 경우를 허용하지 않는다.

```
select name
from background, company
where background.name=~company.name
```

3.1.2. FSQL

본 논문에서 제시하는 FSQL의 정의는 다음과 같다. [1,2,3,4]

```
select A1, A2, ... bound K1
from T1, T2 ...
where [x1 ⊗ Wy1 bound K2] Φ [x2 ⊗ Wy2
bound K3] Φ ...
Ai : Attribute_name, Table_Name.Attribute
      _Name
Ti : Table_Name
Ki : Threshold
```

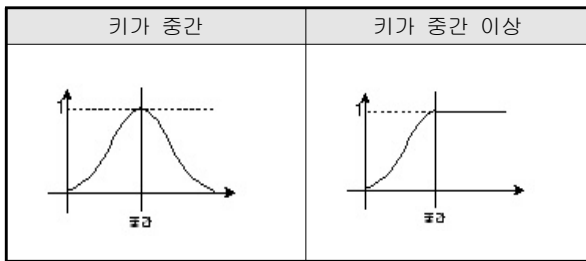
- x_i : Attribute_name for Searching Attribute
- y_i : Value
- W: Reserved Word(very, little, above, below...)
- ⊙ : Comparison Operator(<, ≤, =, ≠, >, ≥, ~)
- ⊕ : Connection Operator(and, or)

연결자여부	fuzzy여부	부호구분	Attribute 구분	Value 구분	정도구분
·(null)→00 ·and→01 ·or→10	·non-fuzzy→00 ·fuzzy→01	=→000 >→001 <→010 >→011 <→100	·Attribute	·Value	·(null)→0 ·very→1

[표 1] FSQL 처리기에서의 input으로 사용되는 Symbol Table

일반 sql을 FSQL로 확장하여, 추가된 부분은 다음과 같다.

- ① bound
bound는 소속 척도 함수의 한계치를 정해준다
- ② 예약어
'very' 혹은 'little'의 예약어를 사용하여 소속척도 함수의 정도를 바꾸어 줄 수 있다. 본 논문에서는 very의 경우 함수의 제곱을 취하고, little의 경우 함수의 제곱근을 취하였다.
또한 '이상의'의 의미를 갖는 'above' 라는 예약어가 있다. 이는 소속 척도 함수를 일정 기준치에서부터 '1'로 만들어 주는 기능을 한다.



[그림 3] 부호에 따른 소속함수 비교

- ③ 비교연산자
비교연산자 중에서 '~'연산자가 있다. 이는 퍼지함수를 의미하며, 퍼지의 의미를 갖고자 하는 대상의 앞에 존재한다. 본 논문에서는 여타 다른 비교연산자는 퍼지 할 수 없도록 한다.
- ④ 결합연산자
결합연산자 and, or는 일반 sql과 동일하나 결합시에 소속척도값의 계산이 요구된다.

(1) BNF

위의 FSQL은 sql형태를 그대로 받아들이며, 몇가지 기능만 추가 하였다. 따라서 이에 대한 BNF형도 sql의 BNF를 그대로 받아들이며 추가되는 사항은 다음과 같다.

- ① 0과 1, 그리고 그 사이의 소수
<numeric literal> ::= 0 | 0[<period>[<unsigned integer>]] | 1
※ <period>와 <unsigned integer>는 sql bnf에 정

- 의되어 있음
- ② ~ 연산자의 추가
<FSQL special character> ::= <SQL special character> | <Tilde>
<Tilde> ::= ~
- ③ very 등의 예약어 추가
<Reserved word> ::= <reserved word> | BOUND | VERY | LITTLE | ABOVE | BELOW | ...
- ④ bound
<Fquery specification> ::= <query specification> | <query specification> <bound clause>
<Fwhere clause> ::= <where clause> | <where clause> <bound clause>
<bound clause> ::= bound <numeric literal>

위에서 정의된 BNF형에 의해 본 시스템에서 사용하는 통일요한 다음과 같은 형태를 갖는다.

```
select name bound 0.8
from background
where age = ~young bound 0.8 and height = 180
```

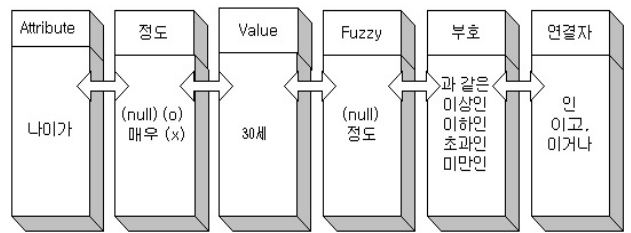
3.2. metaDB 처리

metaDB는 퍼지 질의 처리 시에 참조하는 자체 데이터 베이스로써 퍼지 비교 연산자의 의미나 퍼지 상수 그리고 attribute 간의 유사한 정도를 정의해 둔 데이터베이스이다.

3.2.1. 처리 가능한 음성 질의어

FSQL 처리를 하기 위한 선작업으로 수많은 변형의 형태로 들어오는 음성 질의어의 구문을 파악하여 FSQL 처리기에서 받아들일 대표 attribute와 value에 포함하는 사용 가능할 모든 음성 데이터를 정리하여 metaDB를 구축한다. 그리고 실제 들어온 음성 질의어는 [표 1]과 같이 각각의 attribute를 분류한 Symbol Table의 조합이 FSQL 처리기에서의 Input으로 들어온다.

<예>Attribute="나이" Value="30"을 포함하여 나타낼 수 있는 질의어는 30 가지이다.



[그림 4] Attribute = "나이", Value = "30"을 포함하여 나타낼 수 있는 질의어

3.2.2. 인적사항 attribute별 데이터베이스 분류

인적사항 관련된 데이터를 수집하여 [표 2]에서 보는 바와 같이 attribute 형태를 기준으로 퍼지 여부에 따라 데이터를 분류하였다. [표 2]에서 attribute 형태에 따라 분류한 데이터베이스를 3.2.2.1과 3.2.2.2에서 분류하여 설명한다. 또한 퍼지한 경우에 정확한 값을 대입하기 위하여 임의의 남녀노소 60명을 기준으로 한 통계조사 결과를 사용하였으며 통계청의 객관적 자료를 기준으로 소속함수를 작성하였다.

non-fuzzy	normal form	성별, 연락처, 출신지도, 시/군, 주민등록앞, 뒤, 혈액형, 병역사항, 병역내역, 범죄유무, 범죄내역, 결혼유무, 차량보유유무, 종교
	metadata form	현직장직종, 전공분야, 스포츠, 음악, 영화
fuzzy	continous form	나이, 생년, 월, 키, 몸무게, 결혼기간, 차량보유기간, 경력, 봉급, 토익점수, 토플점수, 총재산
	index form	직위, 최종학력
	matric form	성, 이름, 주소 등

[표 2] 인적사항 Attribute 형태에 따른 데이터베이스의 분류

3.2.2.1. non-fuzzy한 attribute

attribute의 값에 퍼지 성분이 포함되지 않는 데이터들의 분류는 다음과 같다.

① normal form
metaDB의 참조없이 기존의 sql문을 그대로 적용한다.

② metaDB form
상위 레벨의 데이터가 포함하는 하위 데이터를 테이블로 작성하여 metaDB로 참조한다. 예를 들어 “좋아하는 스포츠”가 “수영”일 때 value에 “수중운동”이라는 데이터만 있어도 “좋아하는 스포츠”로 선택되어질 수 있어야 한다.

3.2.2.2. fuzzy한 attribute

또한 attribute의 값에 퍼지 성분이 포함되어지는 데이터들의 분류는 다음과 같고, 세가지 형태로 소속 척도를 구할 수 있다. 퍼지 데이터베이스 모델 중 VAGUE 모델은 각 속성의 정의구역에 대하여 소속척도를 구하기 위해 거리계산법이 할당되는데, 거리계산법은 특정한 두 속성값의 서로 다른 정도를 알 수 있으며, 본 연구에서는 아래와 같이 VAGUE 모델의 변형된 방법을 사용한다.

① continous form
소속척도를 결정하는 방법 중 속성값이 정수나 실수 같은 연속적인 값을 가질 경우에는 함수를 사용하여 소속척도를 구할 수가 있으며, 본 논문에서는 식(1)과 같은 범중형 함수를 사용하였다. [4,5,6,7,8]

$$\mu(x, a, b) = e^{-\frac{1}{a^2}[(x-a)]^2} \quad (x < a)$$

$$= 1(a \leq x < b)$$

$$= e^{-\frac{1}{a^2}[(x-b)]^2} \quad (x > b) \dots \dots \dots (1)$$

식 (1)에서 x는 변수 즉, 관계형 데이터베이스로부터 가져온 비교 값, a와 b는 질의로부터 가져온 찾고자 하는 값, a는 표준 편차이다. 따라서 소속척도가 1이 되는 구간이 한 점일 때에는 일반적인 범중형의 그래프로 나타내어진다. 본 논문에서는 임의의 값인 a를 value의 범위에 따

라 생년과 생월에서는 3, 나이와 키와 몸무게와 차보유기간과 경력과 결혼기간에서는 30, 토익과 토플에서는 300, 총재산에서는 3000로 정한다.

또한 퍼지한 경우에 각 attribute들에 대하여 유사한 정도를 나타내는 소속함수를 구하여 metaDB로 저장한다.

예를 들어 나이의 구간을 나타내는 소속함수는 [표 3]과 같이 여러 형태의 소속함수들이 존재할 수 있다. 다른 attribute들의 퍼지한 value의 소속함수들도 이와 같은 형태의 함수들을 따른다.

case	middle	분포형태
case1		$\mu(x) = e^{-C(X-35)^2}$ (C=0.89* 10 ⁻²)
case2		$\mu(x) = (x/10) - 2$ $= 1$ $= -(x/10) + 5$ ▶ 통계적 수치 표현의 어려움
case3		$\mu(x) = (x/15) - (4/3)$ $= -(x/15) + (10/3)$ ▶ 비교적 소속척도의 값이 적게 나타남
case4		$\mu(x) = e^{-c x-35 }$ (c=0.045) ▶ 비교적 소속척도의 값이 적게 나타남

[표 3] metaDB I - Continous Form

② index form

퍼지 질의에 대한 해당 속성값이 숫자가 아니고 문자일 경우, 즉, “최종학력”이 ‘고졸’ 정도인 사람을 찾을 경우에는 기존의 퍼지 질의어에서는 소속척도를 표에 직접 저장해 놓은 형태를 사용한다. 그러나, 이러한 방법을 사용했을 경우에는 주 데이터베이스에 의존도가 높으며 소속척도들의 삽입 삭제 및 관리가 매우 힘들다. 이에 본 연구에서는 기존의 표를 이용한 방법처럼 소속척도를 직접 표에 저장하지 않고 유사도에 의해 정수화를 시켜 metaDB화함으로써 주 데이터베이스와의 의존도를 줄이며 독립적인 관리가 가능하도록 하였다.

만일, “최종학력”=“고졸”라면 [표 4]의 metaDB를 참조하고 나온 값인 3을 소속함수에 대입하여서 디폴트 값으로 주어진 a=0.8 수준레벨을 이용하여 질의에 해당하는 값을 얻을 수 있다. [4,5,6,7,8]

③ Matric form

성이나 이름의 퍼지한 value를 갖는 데이터들 간에 각 데이터들의 초성, 중성, 종성의 조합으로 유사도를 표현할 수 있다. 한글학회와 그밖의 한국어 관련 논문을 참고하여 먼저 초성, 중성, 종성의 유사도를 나타내는 테이블을 만들었다. 초성의 경우 유사한 자음끼리의 값은 1, 0.8, 0.6,

0.4의 값을 갖고, 중성과 중성의 경우는 유사한 자음 또는 모음끼리의 값은 1, 0.9, 0.8, 0.7 값을 가진다.

예를 들어 성이 “정”인 사람과 “전”인 사람의 유사도는 table에 의해서 초성(1) *중성(1) *종성(0.7) = 0.7의 값을 가진다. 값을 갖는다. 마찬가지로, 주소의 동의 경우는 공간적인 거리로 유사도를 표현하였다.[4,5,6,7,8]

초졸	1
중졸	2
고졸	3
대중퇴	4
대졸	5
대학원졸	6
박사	7

[표 4] metaDB II - Index Form

	ㄱ	ㄲ	ㄴ	ㄷ	ㄸ	ㄹ	ㅁ	ㅂ	ㅃ	ㅅ	ㅆ	ㅇ	ㅈ	ㅊ	ㅋ	ㆁ	ㅅ	ㅇ	
ㄱ	10	8	6	6	4	6	6	6	4	4	4	4	4	4	4	8	4	4	4
ㄲ	8	10	4	4	4	4	4	4	4	4	4	4	4	4	4	8	4	4	4
ㄴ	6	4	10	6	4	6	6	6	4	4	4	4	4	4	4	4	4	4	4
ㄷ	6	4	6	10	8	6	6	6	4	4	4	4	4	4	4	8	4	4	4
ㄸ	4	4	4	8	10	4	4	4	4	4	4	4	4	4	4	8	4	4	4
ㄹ	6	4	6	6	4	10	6	6	4	4	4	4	4	4	4	4	4	4	4
ㅁ	6	4	6	6	4	6	10	6	4	4	4	4	4	4	4	4	4	4	4
ㅂ	6	4	6	6	4	6	10	6	4	4	4	4	4	4	4	4	4	4	4
ㅃ	4	4	4	4	4	4	4	8	10	4	4	4	4	4	4	4	4	4	4
ㅅ	4	4	4	4	4	4	4	4	4	10	8	4	8	4	6	4	4	4	4
ㅆ	4	4	4	4	4	4	4	4	4	8	10	4	4	4	4	4	4	4	4
ㅇ	4	4	4	4	4	4	4	4	4	4	4	10	4	4	4	4	4	4	8
ㅈ	4	4	4	4	4	4	4	4	4	4	8	4	4	10	8	8	4	4	4
ㅊ	4	4	4	4	4	4	4	4	4	4	4	8	10	8	4	4	4	4	4
ㅋ	4	4	4	4	4	4	4	4	4	4	6	8	4	8	8	10	4	4	4
ㆁ	8	8	4	4	4	4	4	4	4	4	4	4	4	4	4	10	6	4	4
ㅅ	4	4	4	8	8	4	4	4	4	4	4	4	4	4	4	4	6	10	4
ㅇ	4	4	4	4	4	4	4	4	4	4	8	8	4	4	4	4	4	4	10

[표 5] metaDB III - Matric Form

3.2.3. 튜플소속척도 구하는 방법

3.3.2.절에서 구한 각 attribute의 소속척도를 가지고 퍼지한 질의에 적합한 최적의 튜플을 선택할 수 있도록 하는 튜플소속척도를 구하는 방법에는 여러 가지가 있지만 본 논문의 FSQL 처리기의 구현에는 모든 영역 소속척도를 튜플 소속척도에 반영해주는 평균치 방법과 N-Root 방법 [4] 을 사용자의 선택(기본은 평균치 방법)에 의해 각각의 방법을 사용할 수 있도록 구현했다.

① 평균치 방법.

평균치 방법은 각 소속척도들을 균일한 중요도로 처리한다.

$$\mu_t = \frac{1}{m} \cdot (\sum_{j=1}^m \mu_j), (1 \leq m \leq n) \dots (2)$$

(μ_t : 튜플소속척도, μ_j : 해당소속척도, m : 소속척도갯수, n : 총 컬럼수)

② N-Root 방법

본 논문의 3.3.2.절에서 제시한 방법으로 구한 각 컬럼의 소속척도들을 곱한 값에 N-제곱근승(N-square root)을 하는 방식으로, 식 (3)과 같다[12].

$$\mu_t = \sqrt[m]{\mu_1 \times \mu_2 \times \dots \times \mu_m} \dots (3)$$

4. FSQL처리기 구현

FSQL 처리기 크게 세 부분으로 구성된다. 먼저 FSQL을 분석하는 fsqlparser와 현재의 상용 데이터베이스를 이용해 입력된 fsql 문장을 처리할 수 있도록 변환해주는 fsql2sql, 마지막으로, 결과 sql을 이용해 쿼리한 데이터를 유저에게 보여주는 fsqlQuery application이다.

첫 번째로 ‘fsqlparser’는 JavaCC를 사용해 구현하였다. JavaCC는 top-down 방식의 LL(k) 파서를 생성하는데, 정규식이나 문자열 같은 구문 명세와 문법 명세(BNF)를 한 파일에 함께 기술하기 때문에 읽기 쉬울뿐만 아니라 유지보수가 쉽다는 이점이 있다. 또한 JavaCC는 순수 자바언어로 구현되었기 때문에 애플리케이션과 동일한 문법이 적용되고, 같은 입력 데이터를 동일하게 사용하므로 parsing time을 줄일 수 있다.

두 번째로 fsql2sql은, sql에서는 사용하지 않는 ‘bound’나 ‘~’ 기호를 사용한 fsql 문장에 의한 쿼리가 가능하도록 fsql 문장을 sql 문장으로 변환하여 주는 기능을 한다. 이미 3절에서 정의된 metaDB를 참조하여 각각의 퍼지 성분의 단어에 대한 퍼지 정도의 값을 결정하고, sql에서는 사용이 불가능한 ‘bound’, ‘~’를 적절한 sql 문장으로 변환한다.”

```

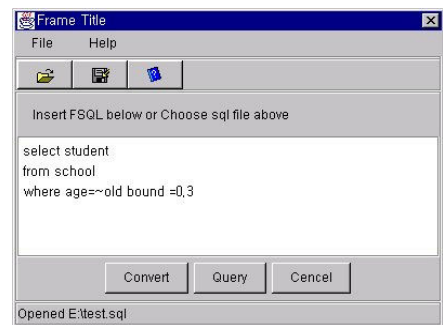
<예 > select manager
from employee
where age=~old bound 0,8
    
```

```

→ select manager
from employee
where 45<=age< 145
    
```

: 나이에 대한 기록이 0세부터 145세까지 있고, metaDB에 속한 old의 0,8에 대한 value가 45세일 때를 가정한다.

마지막으로 fsqlQuery application에서는 앞 단계의 결과인 sql 문장으로 데이터를 쿼리한다. [그림 5]는 fsqlQuery application을 실행시킨 모습이다.



[그림 10] FSQL 처리기의 수행결과

5. 결론 및 향후 연구

데이터베이스 사용의 증가와 데이터베이스 활용의 지능화, 고도화에 대한 욕구는 끊임 없이 계속되고 있다. 이러한 이유로 사용에 제약이 없는 질의어가 필요하게 되었으며, 이에 퍼지 이론을 도입한 퍼지 데이터베이스의 연구가 계속되고 있다.

본 논문에서는 간단한 인적사항 관련 데이터베이스 시스템을 구축하기 위하여 FSQL 처리기를 설계 및 구현하

였다. 즉, 퍼지 질의를 기초로 하여 새롭게 정의한 퍼지 질의어(FSQL)를 사용하여 기존의 관계형 데이터베이스를 기반으로 한 FSQL 처리기를 구현하였다.

기존의 방법들과는 달리 본 논문에서는 확장처리(비교자의 확장, 연결자의 확장)와 MetaDB를 사용함으로써 보다 자연스러운 퍼지 질의의 수행이 가능하다. 또한, 기존에 구축된 관계형 데이터베이스와의 호환하는 것에 주안점을 두었다.

본 연구에서 구현한 FSQL처리기는 제한적인 방법으로 퍼지질의와 자연어처리가 가능한 퍼지질의 시스템의 실현가능성을 보였다. 아직까지는 제한적인 질의어 처리가 이루어지고 있으며, 방대한 량의 데이터베이스에 대해서도 metaDB를 구축하는데 있어서도 다양하고 인공 지능적인 방법이 필요하다. 또한, 인적 사항 데이터베이스를 위한 음성 인터페이스 퍼지질의의 시스템을 구축하는 데에 이 연구의 목적이 있으므로 각 단계의 원활한 인터페이스가 필요하다. 향후에는 이러한 제약점들을 최소화 시킬 것이며, 퍼지 질의의 확장과 정형화에 관한 연구를 계속 할 것이다.

참 고 문 헌

[1] Korth H. F. and Silberschatz, A., Database System Concepts, McGRAWHILL, N. Y., 1991.
 [2] Date, C. J., An Introduction to Database Systems 1, Addison Wesley, 1985.
 [3] Zadeh, L. A., "Fuzzy Sets", *Information Control* 8, pp.338-353, 1965.
 [4] S.C. Park, C.S. Kim and D.S. Kim, Fuzzy Logic and its applications to engineering information sciences, and intelligent systems, KLUWER ACADEMIC PUBLISHERS, 1995, pp. 407-415.
 [5] 이광형, 오길록, 퍼지이론 및 응용 II권
 흥릉과학출판사, 4장, 1991.
 [6] 이도현, 이광형, 김명호, "퍼지 질의를 위한 관계대수의 확장", 정보과학회논문지 제20권, 제2호, 1993.2.
 [7] Buckles, B. P. and Petry, F. E., A Fuzzy Representation of Data for Relational Database, *Fuzzy Sets and Systems* 7, pp.213-226, 1982.
 [8] Lee K. H. and Oh, G. R., Fuzzy Sets and Applications 2, Hongreung Science Publishing Company, 1991.
 [9] 이광규, 전근환, 김흥기, "퍼지 질의어를 이용한 FDBL 모델", Journal of the Research Institute for Computer Science. Vol. 4, No. 1, 1996.2.
 [10] 이석균, "신뢰도와 불확실성을 지원하는 확장 관계형 데이터 모델", 정보과학회논문지 제24권 제4호, 1997.4.
 [11] Buckles, B. P. and Petry, F. E., "Information-Theoretical Characterization of Fuzzy Relational Databases", *IEEE Trans. on Systems, Man, and Cybernetics SMC-13*, 1983, pp.74-77.
 [12] Motro, A. VAGUE, "A User Interface to Relational Databases that Permit Vague Queries", *ACM Trans. on Office Information System* 6, pp.187-214, 1988.
 [13] Sheno, S. and Melton, A., "Proximity Relations in the Fuzzy Relational Database Model", *Fuzzy Sets and Systems* 31, pp.285-296, 1989.
 [14] Zadeh, L. A., "Outline of a New Approach to the Analysis of Complex Systems and Decision Process", *IEEE Trans. on Systems, Man, and Cybernetics*

SMC-3, 1973
 [15] Klir G. J. and Folger, T. A., Fuzzy Sets, Uncertainty, and Information, Prentice-Hall International(UK) Limited, London, 1988.
 [16] Sheno, S., Melton, A. and Fan, L. T., "An Equivalence Classes Model of Fuzzy Relational Database", *Fuzzy Sets and Systems* 38, pp.153-170, 1990.
 [17] Zemankova, M. and Kandel, A., "Implementing Imprecision in Information systems", *Information*



신 세 영

1995-1999 전북대학교 정보통신공학과
 1999-현재 전북대학교 대학원 정보통신학과
 관심분야 : 데이터베이스, 네트워크 프로그래밍, Mobile Computing



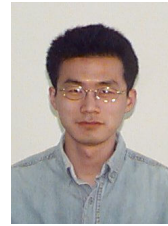
정 은 영

1995-1999 전북대학교 정보통신공학과
 1999-현재 전북대학교 대학원 정보통신학과
 관심분야 : 데이터베이스, 네트워크 프로그래밍, 인터넷 서비스



김 승 권

1991-1999 전북대학교 정보통신공학과
 1999-현재 전북대학교 대학원 정보통신학과
 관심분야 : Mobile Agents, 무선 인터넷(WAP), Fuzzy -SQL/데이터베이스



김 수 영

1993-2000 전북대학교 정보통신공학과
 2000-현재 전북대학교 대학원 정보통신학과
 관심분야 : XML, 네트워크 프로그래밍



박 순 철

1972-1979 인하대 응용물리학과
 1986-1991 미국 루이지애나 주립대 전자계산 박사
 1991-1993 한국 전자통신 연구소 데이터베이스 연구실 과제 책임자
 1993-현재 전자정보공학부 부교수
 관심분야 : 데이터베이스, 네트워크 컴퓨팅, SAN(Storage Area Network),