

## u-쇼핑 시스템을 위한 상황인식적이고 협력적인 질의 시스템 개발\*

권오병  
경희대학교 국제경영학부  
(obkwon@khu.ac.kr)

신명근  
2EIT Inc.  
(mkshin@gsm.kaist.edu)

유비쿼터스 서비스가 여러 분야에서 실용화 단계에 와 있다. 특히 u-쇼핑 분야에서는 사용자의 상황에 맞게 상품 혹은 상점을 유연하게 추천해 주는 개인화된 추천이 필요하다. 그러나 현재의 협력적 질의나 상황인식적 검색은 각각 동적 상황과 모호한 검색을 처리하지 못한다는 한계가 있다. 따라서 본 연구의 목적은 이 두 가지 중요한 문제를 해결하기 위하여 개인화된 상황인식적이고 협동적인 검색방법을 상황적MKAH를 유도방법으로 제안하는 것이다. 본 연구의 가능성을 보이기 위해 CACO라고 하는 프로토타입 시스템을 구현하였으며 국내의 한 대규모 쇼핑몰 도메인에 맞게 적용하였다.

논문접수일 : 2006년 06월

게재 확정일 : 2006년 12월

교신저자 : 권오병

### 1. 서론

유비쿼터스 서비스 시스템이 상용화로의 진전을 보고 있다. 우선 단순한 길 찾기, 위치 기반 광고 등 위치 기반부터 시작하여 상용화가 시작되어 RFID 기반의 SCM이나 원격진료, u-쇼핑 등과 같은 보다 더 기술적 수준이 높고 지능화된 서비스들이 제안되기 시작하였다. 그런데 u-쇼핑과 같은 상황인식적인 서비스에서 주변 상황 요소를 고려하여 선응적이고 개인화된 이러한 적절한 맞춤을 구현하기 다음과 같은 조건들이 만족되어야 한다. 첫째, 상황인식적 서비스는 사용자의 입력을 최소한으로 줄여야 한다. 특히, 정교한 주변환경 기반 시스템은

보다 많은 주변환경 정보를 필요로 한다고 가정할 때, 주변환경 정보를 자동적으로 얻을 수 없다면 사용자는 그러한 시스템을 용납하지 않을 것이다. 따라서, 만약 상황인식적 시스템이 질의 처리기를 포함하고 있다면, 질의 명령어는 사용자의 최소 입력으로만 구성되어야 한다. 둘째, 상황인식적 시스템은 미래 지향적인 서비스를 위해 입력되어진 개인 정보를 통하여 사용자의 사적인 관심사를 분석할 수 있어야 한다. 셋째, 실제 상황에서는 사용자가 조건이 정확한 질의뿐만 아니라 모호한 조건의 질의도 제시할 수 있도록 해야 한다. 모호한 질의는 사용자가 자신이 진정으로 원하는 것을 모를 때 일반적으로 사용된다. 상황인식적 시스템에서 맞춤

\* 본 연구는 21세기 프론티어 연구개발사업의 일환으로 추진되고 있는 정보통신부의 유비쿼터스컴퓨팅 및 네트워크원천기반기술개발사업의 지원에 의한 것임.

질의를 언급하기 위해서는, 주변환경의 인식 및 개인의 관심사를 활용하여 모호한 질의를 정확하게 분석해 내는 것이 반드시 고려되어야 한다.

그 동안 사용자의 모호한 질의를 처리하기 위해 협력적 질의를 제안해 왔다. 모호한 질의에 대한 협력적 질의 처리의 핵심 기능 중 하나는 개념과 개념 사이의 거리를 계산하는 것이다. 개념과 개념 사이의 거리는 의미적으로 인지되는 거리이다. 그러나, 일반 협력적 질의 처리 알고리즘들은 개념 사이의 거리 값에 변화를 주는 사용자 주변 환경의 영향을 거리 계산에 반영하지 못한다. 또한, 현재의 협력적 질의 처리는 개인의 관심사도 고려하지 않는다. 예를 들면, 사용자가 임의의 장소에서 데이터를 할 것이라는 정보를 시스템은 알 수 없을 것이다. 만약 사용자가 질의를 위해 자신의 다음 행동 및 위치를 제공해야 한다면, 그런 시스템들은 사용자로부터 외면당할 것이다. 우리는 이러한 제한을 극복하기 위해 상황인식적 맞춤형 협력적 질의 처리 시스템을 개발할 것이다. 이 시스템은 개인화된 추천시스템으로서도 유용하게 사용될 수 있다.

따라서, 본 논문의 목적은 사용자의 위치나 일정과 같은 주변 정보에 기반 한 안전하고 개인화된 주변환경을 고려하는 질의 처리 방법론을 제안하는 것이다. 우리의 방법론은 질의 처리를 위한 최소한의 입력만을 요구하며 사용자의 모호한 질의와 사적인 관심사를 통해 개인의 취향을 분석한다. 우리는 제안하는 방법론의 가능성을 보이기 위해 CACO(Context-Aware COoperative Query system)라는 모형 시스템을 구현하였으며, 한국의 한 대형 쇼핑몰에 적용하였다.

## 2. 관련연구 : 협력적 질의 처리

데이터베이스 시스템에서 처리하는 자료가 점

점 복잡해 짐에 따라, 사용자가 제시한 질의문과 정확히 일치하는 자료만을 검색하여 반환하는 것은 사용자에게 도움을 주지 못하는 경우가 많다. 따라서 사용자와 협력할 수 있는 데이터베이스 시스템의 중요성이 더욱 높아지고 있다(Barg and Wong 2000; Bernardno et al., 2002; Bosc et al., 2001; Gaasterland 1997; Han et al., 1996; Babcock et al., 2003; Liu and Ng 1998; Wang et al., 2004). 데이터베이스 시스템에서 협력적 질의 처리란 질의 내용과 정확히 일치하는 자료 뿐만 아니라 좀 더 넓은 범위의 자료 또는 근사적인 자료를 검색해 주는 것을 말한다. 협력적 질의 처리는 질의 분석, 질의 유연화, 유연화된 질의에 따른 근사적 자료 제공의 세 단계로 이루어져 있다. 두 번째 단계인 질의 유연화를 수행하기 위해서는, 개념의 추상화와 자료 사이의 정량적인 유사도를 거리로 표현하는 방법 등을 포괄하는 지식 표현 프레임워크가 잘 설정되어야 한다. 여기서 지식 표현 프레임워크는 협력적 질의 응답 처리기의 성능과 특성에 영향을 미치는 가장 중요한 요인이 된다. 지식 표현 기법에 대한 기존의 여러 연구들은 논리 모형(logic models), 의미 거리(semantic distance), 추상화(abstraction) 등을 이용하여 수행되어 왔다.

이와 관련해서 선행된 연구들 중에서도 논리 모형 접근법(Braga et al., 2000; De Sean and Furtado 1998; Gaasterland 1997; Godfrey 1997) 은 이 분야 연구에 커다란 부분을 차지하고 있다. 논리 모형 접근법(the logical model approach)은 데이터 값들 중에서 의미 관계를 표현하고, 보존 제약조건을 표현하기 위하여 일차 논리 조건문(first-order logic predicates)을 사용한다. 그러므로 전체 데이터베이스는 기본적 조건문들의 집합으로 구성된다. 데이터베이스 질의 역시 자유 변수로 상술되는 정보를 알기 위한 조건문의 규칙에 따라 쓰여진다. 여기서 질의는

충돌 해결(conflict resolution)과 추론 기법(inference mechanism)을 통하여 응답된다. 질의 완화는 완결성 제약(integrity constraints)들을 통합시키면서 실행된다. 그러나 이러한 접근 방법은 체계적인 구성이 미흡하고, 질의 유연화 과정을 보여주지 않는 한계가 있다. 그러므로 논리적 모형 접근법은 큰 규모의 시스템에 적용시키기에는 적합하지 않다.

추상화 접근법(abstraction approach) (Chu et al., 1996; Huh and Lee 2001; Vrbsky and Liu 1993)은 데이터 값들 간의 의미 관계를 표현하는데 효과적인 방법이라고 평가 받아온 데이터 추상화 방법(Abiteboul and Duschka 1998; Abiteboul et al., 2002)을 사용한다. 협력적 질의 응답에서 데이터 추상화와 같은 기법은 질의 완화를 위하여 데이터 값들을 연관 지어 주는데 유용한 방법이기도 하다. 여러 데이터 값들 중에서 서로 관련 있는 데이터 값들끼리 구성해주는 추상화 개념을 구체적으로 실현하기 위하여 타입 추상화 계층(type abstraction hierarchy) (Chu et al., 1996)에서는 포함의 개념(notions of subsumption), 구성의 개념(notions of composition), 그리고 추상화의 개념(notions of abstraction)을 설명하였고, 다중 레벨지식 추상화(multi-level knowledge abstraction)를 바탕으로 타입 계층화를 설명하기 위한 완전한 관점을 제시하였다. 타입 추상화 계층은 협력적 결과를 유도하기 위하여 지식 레벨 차이에 따라 질의 다시 쓰기(query rewrite)와 추론을 손쉽게 하기 위한 주제 연관(subject association)을 사용하였다.

의미 거리 접근법(semantic distance approach)은 데이터 값들간의 유사도를 표현하기 위하여 의미 거리 개념을 사용한다(Cuzzocrea and Matrangolo 2004; Ichikawa and Hirakawa 1986; Motro 1988, Motro 1996; Palpanas 2001). 데이터 집합 안의 모든 데이터 값들의 순서쌍은 의미 거리를 갖는다고

가정되므로(Motro, 1990) 의미 거리 접근법은 질의 유연화에 간단하면서도 효과적인 방법을 제공하고, 순위를 매겨 의미 거리 값을 기준으로 정렬해준 결과를 제시해준다. 이렇게 순위가 매겨진 결과는 사용자가 그 결과에서 유용한 정보를 찾아낼 수 있는데 도움을 준다. 문자열 도메인 자료에 대해서는 두 데이터 값들 간의 거리를 표로 정리하여 보 관하며, 표의 크기가 커질수록 생성 및 유지 비용이 기하급수적으로 커지는 단점이 있다. 의미 거리 접근법과 추상화 접근법을 통합하려는 시도도 이루어졌다. 추상화 접근법에서는 숫자 도메인에 대한 추상화 방법(Chu et al., 1996)과 MKAH(Shin et al., 2007) 등이 제안되었다. 그러나, 이러한 방법들 모두 유동적인 주변환경과 개인화 특성을 지원하는 질의 처리는 하지 못하였다.

기존의 협력적 질의 처리 시스템들은 주변환경 인식과 사생활이 보호되는 개인화의 측면을 고려하지 않는 경우가 대부분이었다. 상황인식적 컴퓨팅은 주변환경에 기반 한 IT 서비스를 제공함에 있어서, 유용한 주변환경 정보를 자동으로 얻어서 사용자의 수동 입력을 최소화하며 계산 환경을 감추는 것을 목표로 한다. 또한, 산만한 결과값이 나오는 것을 최소화하기 위해서, 퍼제형 시스템(pervasive system)과 같은 개인화 시스템은 반드시 주변 환경을 고려해야 한다(Anhalt et al., 2001). 이러한 특징들은 맞춤 개인화의 현실화를 가능하게 해주는 것으로 인정되었다.

### 3. 상황인식적이고 협력적인 질의

#### 3.1 개요

SQL명령문으로 이루어지는 전통적인 검색은

정확한 검색만이 가능하다. 예를 들어 테이블의 속성으로 사전에 정의되지 않는 속성에 대해서는 결과가 나오지 않거나 모호한 값의 검색은 불가능하다. 그에 비하여 협력적 질의는 정확한 정보뿐 아니라 적절한 정보나 심지어는 인접 정보(approximate information)를 제공할 수 있다는 장점이 있다. 하지만 협력적 질의는 여전히 사용자의 상황을 인식하지 못함으로 정적인 검색만을 가능하게 할 뿐이다.

한편 상황인식 기술에 대한 연구가 진행되면서 어떤 상황에서는 검색되고 어떤 상황에서는 검색되지 말아야 할 검색 문제에 대한 해결 방법이 가능하게 되었다. 예를 들어서 사용자의 위치가 확인이 되면 그 위치에 적절하지 않은 검색 결과는 비록 사용자가 그것을 인지하지 못했다고 하더라도 제외할 수 있을 것이다. 이러한 상황인식적인 검색은 검색결과와 적절성을 향상시킬 것으로 기대된다. 특히 사용자의 프로필이라고 하는 내부 상황정보(internal context)는 웹으로부터 검색되는 웹 페이지의 정확성을 높이는 데 사용될 수 있다(Storey et al., 2004). 하지만 이러한 일련의 상황인식 검색은 모호한 검색을 지원하는 것은 아니다. 이러한 한계점은 검색 시스템이 실제 사용되는데 있어서 상황 정보가 정확하게 정의되지 않은 문제 영역에서는 잘 작동하지 않는다는 문제를 야기한다. 따라서 본 연구에서는 상황인식 검색과 협력적 질의의 장점을 결합한 상황인식적인 협력적 질의 시스템을 고려하였다.

### 3.2 MKAH를 활용한 개념간 거리 계산

본 연구의 검색 방법을 제안하기 위해서 먼저 개념간 거리를 계산하는 방법을 MKAH(Metricized Knowledge Abstraction Hierarchy)방법을 근간으로 하여 제안하였다. 여기서의 개념이란 개인의 프로필, 선호도, 검색 대상, 상황정보 등이 모두 포함

된다. 예를 들어 검색 대상과 관련한 카테고리인 음식점과 영화관은 개념적으로 얼마나 가까운지, 또한 이 개념간 거리는 음식점과 카페라고 하는 개념 사이의 거리에 비하여 먼 것인지 아닌지를 계산하는 것이다. 이는 음식점을 찾는 검색이 정확한 검색이 아니라 모호한 검색을 목적으로 이루어질 경우 경우에 따라서는 음식점이 아닌 카테고리에서도 검색을 실시할 경우 보다 더 정확성을 높이기 위해 보다 더 가까운 개념에 속하는 카테고리에서도 검색을 실시하여 검색 상의 융통성을 기하기 위함이다.

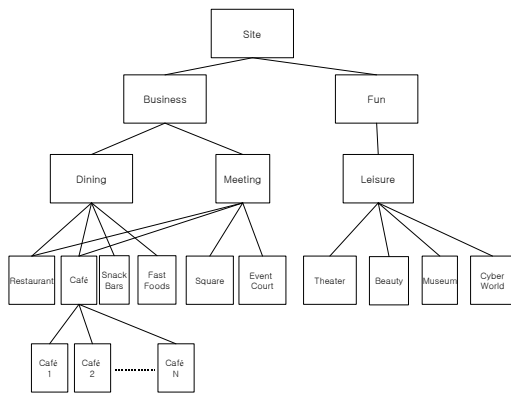
이를 위해 먼저 개념간 거리 측정을 위한 측정치와 최소 거리를 제안하여 개념간 거리의 요구 조건을 만족하는 거리를 정하려고 한다. 이를 위해 특정 영역에 소속한 개념들의 계층을 [그림 1]과 같이 표현하고 거리를 계산할 수 있는 MKAH를 활용하려고 한다(Shin et al., 2007).

먼저 모든 정적인 MKAH의 계층 내에서의 개념간 거리는 다음 [그림 1]과 같이 확보된다.

<p>단계 1-1 : 특정 도메인에 존재하는 노드(예 : 쇼핑몰 도메인인 경우 입점업체)를 인식한다.</p> <p>단계 1-2 : 특정 도메인 내에서 사용자가 가질 수 있는 활동을 인식한다.</p> <p>단계 1-3 : 개념 값들의 추상적 계층인 값 추상 계층(value abstraction hierarchy)을 작성한다.</p> <p>단계 1-4 : 값 추상 계층내의 각 노드간 개념 거리를 다음과 같이 정의한다.</p> $cd(node\_value, Activity = activity\_value)$ <p>단계 1-5 : 단계 3에서 선언된 개념간 거리를 가지고 임의의 두 노드간의 정적인 개념간 거리를 계산한다. 이때 정적인 개념간 거리는 다음과 같은 공식을 따른다.</p> $cd(nodeA,nodeB) = (d_{y,1} + d_{y,2} + \dots + d_{y,N}) / N$ <p>단, 여기서 <math>d_{i,j}</math>는 <math>i</math>번째 노드의 <math>j</math>번째 활동에 대한 개념 거리이다.</p> <p>단계 1-6 : 단계 4에서 계산된 정적인 개념간 거리를 활용하여 개념간 거리 행렬을 작성한다.</p>
---

[그림 1] 정적인 개념간 거리 및 MKAH 도출 방법

예를 들어 쇼핑몰을 도메인으로 하고 그 안에는 Restaurant, Café, SnackBars, FastFoods, Square, EventCourt, Theater, Beauty, Museum 및 Cyber World 등 열 가지의 종류의 노드가 있다고 하자. 그리고 각 노드별로는 인스턴스에 해당하는 각 입점 업체가 있게 될 것이다. 그리고 [그림 2]에서 나온 대로 사용자들이 이 도메인에서 가지게 되는 활동이 식사(dining)와 여가(leisure) 그리고 회의(meeting)의 세 가지라고 하자. 이를 근거로 하여 다음 [그림 1]과 같은 값 추상 계층을 가질 수 있을 것이다.



[그림 2] 값 추상 계층 예

이때 각 노드의 활동별 개념 거리가 다음 [그림 3]과 같다고 하자. 이 자료는 실제로 사용자에게 질문하여 획득하게 되는데, 입력을 하지 않으면 디폴트 값을 입력하게 되고, 나중에 사용하면서 적응적으로 보정하게 된다.

개념거리가 구해지게 되면 각 노드별 개념간 거리를 구하게 된다. 개념간 거리는 단계 1-5의 공식을 따르게 되어 다음과 같이 자동적으로 계산된다.

$$cd( Restaurants, Square) = \sqrt{(0.4 - 0.9)^2 + (0.1 - 0.2)^2 + (0.5 - 0.3)^2} = 0.53$$

node	activity	userID	distance
restaurant	meeting	obkwon	0,4
restaurant	dining	obkwon	0,1
restaurant	leisure	obkwon	0,5
square	dining	obkwon	0,9
square	meeting	obkwon	0,2
square	leisure	obkwon	0,3
theater	dining	obkwon	0,2
theater	meeting	obkwon	0,7
theater	leisure	obkwon	0,6
museum	dining	obkwon	0,2
museum	meeting	obkwon	0,6
museum	leisure	obkwon	0,5
cyberworld	dining	obkwon	0,1
cyberworld	meeting	obkwon	0,3
cyberworld	leisure	obkwon	0,9
snackbar	dining	obkwon	0,85
snackbar	meeting	obkwon	0,7
snackbar	leisure	obkwon	0,4
fastfoods	dining	obkwon	0,9
fastfoods	meeting	obkwon	0,65

[그림 3] 개념 거리 입력 예

그 결과의 일부를 <표 1>에서 보았다. <표 1>의 결과를 통하여 정적 MKAH를 도출하게 된다.

<표 1> 개념간 거리 도출 예

	Restaurant	Square	Theater
Restaurant	0	0.53	0.45
Square		0	0.50
Theater			0

### 3.3 상황적 MKAH

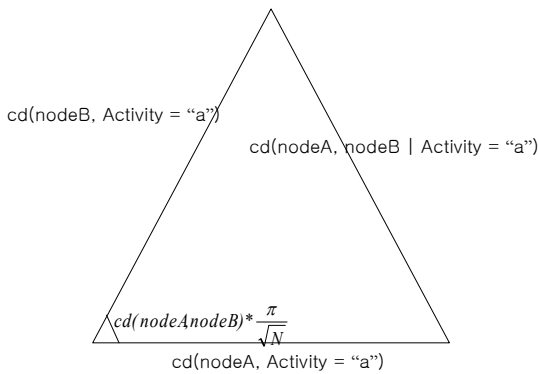
상황적(contextual) MKAH를 구하는 것은 정적 개념간 거리를 반영한 정적MKAH에서 출발한다. 여기서 상황적이라고 하는 것은 사용자의 곧 다가올 활동(upcoming activity)이 된다. 실제로 사용자의 단말기에 들어있는 스케줄 정보를 POOM이라고 하는 Java API를 활용하여 받아와 상황인식을 하는 것으로 하였다. 이때 <표 1>과 [그림 3]에 나온 정보를 활용하여  $cd(\text{nodeA}, \text{nodeB} | \text{Activity} = "a")$  를 도출하게 된다. 이는 노드A와 노드B가

a라고 하는 활동에 대해서 얼마나 개념적으로 가까운지를 구하는 것이며 Euclidean distance of any two nodes,  $cd(nodeA, Activity = "a")$ 와  $cd(nodeB, Activity = "a")$  사이의 변형된 유클리디언 거리 (amended Euclidean distance)로 구하게 된다. 즉,  $cd(nodeA, Activity = "a")$ 와  $cd(nodeB, Activity = "a")$ 의 직접적인 거리가 아니라 저적인 개념간 거리인  $cd(nodeB, nodeA)$ 에 거리가 비례하는 수정적 거리가 된다. 이는 정적인 거리 자체를 중시하기 위해서 이다.

본 논문에서는 이를 다음 [그림 4]와 같이 구하였다. 단 여기서 N은 고려된 활동의 수인 동시에 개념간 거리가 허용 가능한 최대치이기도 하다. 본 논문의 예제에서 보면 활동의 수가 3이며, 이때 허용 가능한 개념간 거리의 최대치는,

$$cd(nodeA, nodeB) = \sqrt{(1-0)^2 + (1-0)^2 + (1-0)^2} = \sqrt{3}$$

이므로 이때 180도를 이루게 하기 위하여 개념간 거리에  $\pi/\sqrt{N}$ 을 곱한 것이다.



[그림 4] 상황적 개념간 거리 도출 개념

상황적 개념간 거리를 통하여 상황적 MKAH를 도출하는 방법을 [그림 5]에 요약하였다.

단계 II-1 : 임의의 두 노드에 대해서 상황적 개념간 거리를 다음과 같이 구한다.

$$cd(nodeA, nodeB | Activity = 'a') =$$

$$cd(nodeB, Activity = 'a') \sin(cd(nodeA, nodeB) * \frac{\pi}{\sqrt{N}})$$

단계 II-2 : 상황적 개념간 거리를 통하여 상황적 MKAH를 도출한다.

단계 II-3 : 개인화된 검색을 위해 특정 노드 인스턴스의 위치인 (x, y)를 함수 f로 구한다.

$$\alpha \xrightarrow{f} (x, y)$$

단계 II-4 : (x, y)와 사용자의 현재 위치를 가지고 임의의 노드 인스턴스에 대한 그 사용자의 인지적 거리를 다음과 같이 추정한다.

$pd(\alpha) = PhysicalDistance^{g(p)}$ , 단, 여기서  $g(p)$ 는 각각 상황적 압력에 대한 주관적 수치인 p에 대한 민감도 함수이다. 주관적 민감도 수치는 5점 척도로 구한다.

단계 II-5 : 각각의 노드 인스턴스에 대해 사용자의 인지적 거리를 반영한 상황적 개념간 거리에 대한 특정 사용자의 선호도는 다음과 같이 구한다.

$$\forall \alpha \in nodeA, \alpha = g(p)$$

[그림 5] 상황적 MKAH도출

#### 4. 구현

본 연구에서 제안한 상황인식적이고 협동적인 검색 방법이 현실적인 시스템에 적합한 지를 알아보기 위해 CACO(Context-Aware COoperative Query system)라고 하는 프로토타입 시스템을 개발하고 국내의 실제적인 한 쇼핑몰인 C쇼핑몰을 대상으로 시스템을 구축하였다. 이 C쇼핑몰은 한국 최대 규모이며 하루 약 15만 명에서 25만 명 정도가 이용하고 있다. 이러한 이용객들이 쾌적하게 쇼핑을 하게 하고자 폭이 18미터인 복도가 이어져 있으며 연장 600여 미터가 넘는다. C쇼핑몰의 주요 성공 요인은 하루에 방문하는 이용객의 수의 증가에 있는데, 이 자료를 기초로 하여 입점업체들과 계약을 하기 때문이다. 그러나 C쇼핑몰은 인근에 지하철

과 호텔, 업무용 빌딩 등 복잡한 경로로 사람들이 접근하며 또한 다양한 목적으로 접근하기도 한다. 따라서 어떤 사람이 이 쇼핑몰에 들어와서 어떤 시설물을 활용하고 나갈 지에 대해서 예측하기란 쉽지 않다. 더욱이 방문 목적도 공적인 목적부터 매우 사적인 목적까지 존재하여 장소 검색에 있어서 사생활 침해도 우려된다. 따라서 사용자들에게 보다 더 효율적이면서도 수용성이 높은 장소 검색 서비스를 제공할 필요성이 높은 상태이다.

한편 C쇼핑몰을 이용하는 사람들은 자신이 원하는 상점 또는 장소가 어디에 있는지를 찾는데 매우 곤란함을 표하고 있다. 그 이유는 안내 데스크에 접근하기가 쉽지 않고 설치되어 있는 안내판이 매우 까다롭게 되어 있기 때문이다. 따라서 장소 검색을 쉽게 받는 것에 대한 요구가 많다.

한편 C쇼핑몰은 소위 물족이라고 하는 회원을 운영하고 있다. 이를 위해 회원 가입 신청자들은 자신의 간단한 프로필 정보를 제공하고 유료로 회원카드를 발급 받으며 그 카드를 통해 여러 가지 혜택을 받을 수 있게 되어 있다. 이것의 의미는 장소 검색 시스템을 구축함에 있어 사용자가 사전에 미리 입력해야 하는 수고로움을 덜 수 있다는 의미이다. 또한 C쇼핑몰은 무선 인터넷 접속이 용이하여 무선랜을 통하여 사용자의 현재 상황을 어느 정도 인식할 수 있다는 조건이 구비되어 있다. 이를 통해 상황인식적인 장소 검색이 가능하다. 그리고 C쇼핑몰에는 이미 입점 업체들에 대한 나름대로의 카테고리화가 되어 있다. 따라서 상점관련 개념들을 체계적으로 정리하는 데 사전적 정보가 준비되어 있는 셈이다.

결국 C쇼핑몰은 규모면이나 복잡도 면에서 장소 검색 시스템이 매우 필요한 상황이며, 회원 정보의 사전 확보, 무선 연결성, 그리고 입점 업체에 대한 체계적 정보라고 하는 세가지 이유로 검색

시스템을 구현하고 운영하기 위한 실현 가능성을 충분히 가지고 있는 대상으로 평가되었다.

CACO는 사용자 프로필, 상황정보의 관점에서 본 주어진 조건 하에서 상점 추천을 유연하게 하기 위한 유비쿼터스 서비스 플랫폼이다. CACO는 사용자의 현재 위치와 그 사람의 현재 주어진 활동 정보 등 상황 정보를 자동으로 인식하고 사용자 에이전트와 상점 추천 에이전트가 포함된 다중 에이전트 방식을 통하여 서비스를 제공하는 것으로 설계되었다. 에이전트와 사용자 인터페이스 구현은 각각 JDK1.4버전과 JSP를 사용하였으며, 각종 정보 저장은 JDBC 연결방식을 따랐다. 사용자와 서비스에 대한 온톨로지는 OWL-Lite으로 표현하고 Jena API를 통하여 파싱 및 검색을 하도록 구현하였다. 한편 사용자와 특정 상점 사이의 거리를 계산하는 서비스는 DistanceService 라고 하는 Apache SOAP 2.0 서버 프로그램의 형태로 구현하였다. DistanceService가 참조하는 데이터베이스의 예를 [그림 6]으로 예시하였다.

no	node	instance	x	y
1	restaurant	ackkabel	74,15	44,8
8	square	asem_sea_food	71,51	54,01
11	restaurant	bennigans	15,73	51,58
12	restaurant	bruschetta_pix_café	86,82	44,2
13	fastfoods	burger_king	37,29	27,52
17	square	chinez_2go	71,51	54,01
18	snackbar	chowon_ginseng_chicken_soup	54,68	42,62
19	snackbar	chung_king	82,66	41,77
20	restaurant	coz	85,05	54,86
21	snackbar	crown_bakery	15,73	46,71
22	square	danbi	71,51	54,01
23	restaurant	deep_blue	85	32
24	snackbar	domodazzi	32,49	36,27
25	square	domoya	71,51	54,01
26	snackbar	dongchunhong	37,85	25,61
27	square	dongkyungam	71,51	54,01
28	cyberworld	game_champ	62,3	47,52
29	square	ganemochi	71,51	54,01
31	snackbar	hanabi	42,93	19,69
32	snackbar	heosuabi	47,13	28,15
34	restaurant	jackys_kitchen	11,43	51,4
35	snackbar	jaedong	69,07	49,95
36	square	jangwon	71,51	54,01
37	square	jeonju_bibimbab1	71,51	54,01
38	square	jeonju_bibimbab2	28,65	33,2
40	restaurant	jugjug_beer	49,5	41,45
41	fastfoods	KFC1	10,2	51,36
42	fastfoods	KFC2	28,65	33,2

[그림 6] DistanceService가 참조하는 상점별 위치 좌표 파일

CACO시스템의 사용자 인터페이스는 [그림 8]과 같다. 좌측 스크린은 사용자 인터페이스인데 협력적 질의를 위하여 다음과 같이 네 가지 종류의 입력을 가능하게 하였다.

• **검색하고자 하는 노드**

본인이 찾아가려고 하는 상점의 종류(노드)를 지정할 수 있으며, 한편 지정을 하지 않고 찾게 할 수도 있다.

• **추천하려는 상점까지의 거리에 대한 사용자의 주관적 민감도**

상점까지 찾아가는데 소요되는 시간을 거리에 비례한다고 가정하고 그만한 시간을 들여서 방문하는 것에 대해서 평소에 얼마나 압박을 느끼는지에 대해 5점 척도로 답하게 한다. 그리고 이 점수는 사용자 정보로 저장되었다가 재차 검색을 하려는 경우 기본값으로 사용자에게 제공된다. 사전에 언급한 대로 주관적 민감도는 심리학의 뉴로티시즘에 근간을 둔 개념이다(Schuler, 1980; Lazarus,

1984; Gonzales et al., 2001).

• **다가올 활동**

다가올 활동에 대해서 선택케 하는 것이며 사용자는 특정한 한 활동을 선택할 수도 있고 아니면 자신의 스케줄대로('as my schedule') 검색하라고 명령할 수도 있다. 스케줄은 사용자에게 의하여 PDA와 같은 모바일 단말기에 전자지갑(eWallet)이라고 하는 사용자 에이전트가 관리하는 정보저장소에 저장 가능하데, 저장된 자료의 예는 다음 [그림 7]과 같다. 사용자 에이전트는 현재 시각에

start_time	end_time	activity	schedule_name
08	09	dining	breakfast
09	11	meeting	buisness meeting
16	18	dating	date with her
18	20	dining	dinner
20	22	rest	for good night
11	12	shopping	for my children
14	16	leisure	gaming!
22	24	rest	good night!
12	14	dining	lunch

[그림 7] eWallet 상의 개인 스케줄 예



[그림 8] 모바일 단말기로 나타나는 CACO 실제 화면



맞는 활동을 검색하여 가져온다.

#### • 정확한 검색 요청 여부

정확한 검색을 할 것인지 아니면 모호한 검색을 할 것인지를 사용자가 지정할 수 있다. 기본적으로는 모호한 검색으로 지정되어 있다. 예를 들어 노드를 Restaurant로 결정하고 모호한 검색을 하게 되면 Restaurant과 연결된 개념 최대치인 dining에 의하여서만 검색을 하지 않고 차상위 개념들인 meeting이나 leisure에 의하여도 검색을 하게 된다. 물론 각 활동에 대해서 노드간 개념거리에 비례하여 고려하는 다른 노드의 인스턴스값의 선호도치가 낮아질 것이다.

위의 네 가지 요인에 대해 선택을 하고 나면 모든 노드 인스턴스인 상점들에 대해서 상황적 개념 거리를 구하고 이것을 다시 인지적 거리로 보정하여 최종 선호도치가 구해지며, 이 중에서 상위 몇 가지를 사용자의 단말기로 보내어 디스플레이하게 된다. 그 예가 [그림 8]의 가운데 스크린이다. 그리고 사용자가 그 중에서 관심이 있는 어떤 상점을 선택하게 되면 그 상점에 대한 광고 정보와 위치 안내가 제공된다. 이러한 상점 정보는 웹서비스의 형태로 제공 가능하며, 정보들은 온톨로지 파일 혹은 WSDL의 형태로 유지 가능하다.

## 5. 결론

본 연구에서는 보다 수용성이 높은 개인화 된 검색을 위해 상황인식적이고 협동적인 검색 방법을 상황적인 개념거리 및 상황적MKAH 정보에 근거를 두어 제안하고, 이 방법이 상황인식적 개인화 서비스에 어떻게 적용될 수 있는지를 밝혔다. 특히 최근의 모바일 시스템 혹은 유비쿼터스 시스

템이 점차로 사용자의 현재 상황을 더욱 더 많이 그리고 자동화된 방법으로 인식하게 하고 있기 때문에 이러한 방법은 개인화 된 지능형 시스템 구현에 시의 적절할 것으로 보인다. 그 중에서도 추천 서비스는 가장 대표적인 서비스 분야가 되고 있으며 개인화된 검색 시스템이 가장 적합한 분야로 생각된다. 현재의 협력적 질의 시스템은 상황인식 기능이 부족하고 상황인식 시스템에서는 모호한 검색을 지원하지 못했는데, 본 연구에서는 이를 노드간 개념거리라고 하는 방법을 사용하여 혼합 활용함으로써 더욱 개인화된 추천을 가능하게 하였다는 것이 가장 주된 공헌이다.

본 연구에서 제안한 방법은 다음과 같은 이유에서 독창적이라고 보여진다. 첫번째로 현존 검색 시스템이 단어나 개념 유사성을 기반으로 선정하는 것과 비교해 볼 때 우리의 방법은 사용자의 상황에 따라 개념간 유사도가 동적으로 변동하는 것을 허용한다는 점이다. 둘째로, 활동이나 노드에 대해서 정확하게 기입하지 않더라도 유연하게 유사한 검색을 수행할 수 있다는 점이다. 이러한 두 가지 특징을 통해서 검색 시스템의 유용성과 따라서 사용자 수용성을 더욱 높일 것으로 기대한다.

하지만 서비스 온톨로지나 개인 온톨로지의 크기가 더해지면서 규모성의 이슈가 대두될 것으로 보이며 이 때문에 개인이 원하는 시간 내에 반응하지 못할 가능성이 있어 시스템 사용의 현실성이 떨어질 수 있다는 사실에 대해 아직 확연하게 증명하지 못했다. 일반적으로 지식베이스가 증가하면 검색 성능은 지식기반 시스템에 있어서 매우 중요한 성공요인이 된다(O'Leary, 1998). 더욱이 완전한 구현이 아직 종료되지 않았으며 이 부분에 대해서는 추후 연구 방향으로 삼고 후속 연구를 진행 중이다.

## 참고문헌

- [1] Abiteboul, S. and Duschka, O. M. "Complexity of Answering Queries Using Materialized Views," *Proceedings of the 17th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*, Seattle, Washington, (1998), 254-263.
- [2] Abiteboul, S., Benjelloun, O. and Milo, T. (2002). "Web services and data integration," *Proceedings of the 3rd International Conference on Web Information Systems Engineering (WISE 2002)*, Singapore, 3-6.
- [3] Anhalt, J., Smailagic, A., Siewiorek, D.P., Gemperle, F., Salber, D., Weber, S., Beck, J., Jennings, J. "Toward Context-Aware Computing: Experiences and Lessons", *IEEE Intelligent Systems*, Vol.16, No.3(2001), 38-46.
- [4] Babcock, B. Chaudhuri, S. and Das, G. "Dynamic Sample Selection for Approximate Query Processing", *Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data*, San Diego, California, USA, (2003), 539-550.
- [5] Barg, M. and Wong, R. K. "A Multi-Agent Architecture for Cooperative Query Answering", *Proceedings of the 33rd Hawaii International Conference on System Sciences*, Maui, Hawaii, 2000.
- [6] Bernardino, J. Furtado, P. and Madeira, H. "Approximate Query Answering Using Data Warehouse Striping," *Journal of Intelligent Information Systems*, Vol.19, No.2(2002), 145-167.
- [7] Bosc, P., Motro, A. and Pasi, G. "Report on The fourth International Conference on Flexible Query Answering systems," *SIGMOD Record*, Vol.30, No.1(2001), 66-69.
- [8] Braga, J. L., Laender, A. H. F. and Ramos, C. V. "A Knowledge-Based Approach to Cooperative Relational Database Querying", *International Journal of Pattern Recognition and Artificial Intelligence*, Vol.14(2000), 73-90.
- [9] Chu, W., Yang, H. and Chow, G. "A Cooperative Database System (CoBase) for Query Relaxation", *Proceedings of the 3rd International Conference on Artificial Intelligence Planning Systems*, Edinburgh, 1996.
- [10] Cuzzocrea, A. and Matrangolo, U. "Analytical Synopses for Approximate Query Answering in OLAP Environments", *Proceedings of the 15th International Conference on Database and expert systems applications*, Zaragoza, Spain, (2004), 359-370.
- [11] De Sean, G. J. and Furtado, A. Z. "Towards a Cooperative Question-Answering Model", *Lecture notes in computer science*, Vol.1495 (1998), 354-365.
- [12] Gaasterland, T. "Cooperative Answering through Controlled Query Relaxation", *IEEE Expert*, Vol.12, No.5(1997), 48-59.
- [13] Godfrey, P. "Minimization in Cooperative Response to Failing Database Queries", *International Journal of Cooperative Information Systems*, Vol.6, No.2(1997), 95-149.
- [14] Gonzales, N. A., Tein, J., Sandler, I.N., and Friedman, R.J. "On the limits of coping: interaction between stress and coping for inner-city adolescents", *Journal of Adolescent Research*, Vol.16, No.4(2001), 372-395.
- [15] Han, J., Huang, Y. and Cercone, N. "Intelligent Query Answering by Knowledge Discovery Techniques", *IEEE transactions on knowledge and data engineering*, Vol.8, No.3

- (1996), 373-390.
- [16] Huh, S. Y., Lee, J. W. "Providing Approximate Answers Using a Knowledge Abstraction Database", *Journal of Database Management*, Vol.32, No.2(2001), 469-484.
- [17] Ichikawa, T. and Hirakawa, M. "ARES: A Relational Database with the Capability of Performing Flexible Interpretation of Queries", *IEEE Transaction on Software Engineering*, Vol.12, No.5(1986), 624-634.
- [18] Lazarus, R. and Folkman, S. *Stress, appraisal, and Coping*, New York: Springer, 1984.
- [19] Liu, Q. and Ng, P. A. "A Query Generalizer for Providing Cooperative Responses in an Office Document System", *Data and Knowledge Engineering*, Vol.27, No.2(1998), 177-205.
- [20] Motro, A. "VAGUE: A User Interface to Relational Databases that Permits Vague Queries", *ACM Transactions on Office Information Systems*, Vol.6, No.3(1988), 187-214.
- [21] Motro, A. "Cooperative Database Systems," *International Journal of Intelligent Systems*, Vol.11, No.10(1996), 717-732.
- [22] O'Leary, E. D. "Using AI in Knowledge Management: Knowledge Management and Ontologies," *IEEE Intelligent Systems*, Vol.13, No.3(1998), 34-39.
- [23] Palpanas, T. and Koudas, N. "Entropy Based Approximate Querying and Exploration of Datacubes", *Proceedings of the Scientific and Statistical DataBase Management*, Fairfax, VA, USA, (2001), 81-90.
- [24] Schuler, R. Definition and Conceptualization of Stress in *Organizations. Organizational Behavior and Human Performance*, 25, (1980), 184-215.
- [25] Shin, M.K., Huh, S.Y. Lee, W. "Providing Ranked Cooperative Query Answers Using the Metricized Knowledge Abstraction Hierarchy", *Expert Systems with Application*, Vol.32, No.2(2007), 469-484.
- [26] Storey, V.C., Sugumaran, V. and Burton-Jones, A. "The Role of User Profiles in Context-Aware Query Processing for the Semantic Web", *Lecture Notes in Computer Science*, (Eds.) Farid Meziane, Elisabeth Metais Springer-Verlag GmbH, Vol.3136 (2004), 51-63.
- [27] Vrbsky, S. V. and Liu, W. S. "APPROXIMATE-A Query Processor that Produces Monotonically Improving Approximate Answers," *IEEE Transactions on Knowledge and Data Engineering*, Vol.5, No.6(1993), 1056-1068.

Abstract

## A Context-Aware Cooperative Query for u-Shopping Systems

Ohbyung Kwon\* · Myung Keun Shin\*\*

Ubiquitous computing technologies become mature enough to be applied in acceptable ubiquitous services. In particular, in u-shopping area, personalized recommender systems which automatically collect the nomadic user-related context data and then provide them with products or shops in a flexible manner. However, legacy cooperative queries and context-aware queries so far do not come up with dynamically changing situations and ambiguous query commands, respectively. Hence, The purpose of this paper is to propose a personalized context-aware cooperative query that supports a multi-level data abstraction hierarchy and conceptual distance metric among node instances, while considering the user's context data. To show the feasibility of the methodology proposed in this paper, we have implemented a prototype system, CACO, in the area of site search in a large-scale shopping mall.

**Key words** : Context-aware Computing; Cooperative Query; Abstraction Hierarchy; Concept Distance

---

\* School of International Management, Kyunghee University

\*\* 2EIT Inc.