

상황 온톨로지를 이용한 동적 의사결정시스템

김현우

한국과학기술원 EEWS 대학원
(kimhyunu@kaist.ac.kr)

손미애

성균관대학교 공과대학 시스템경영공학과
(myesohn@skku.edu)

이현정

서강대학교 경영대학
(hjlee5249@yahoo.com)

.....

본 연구는 사용자의 정적, 외부환경과 연관된 동적 상황정보와 사회적 관계와 연관된 개인적 상황정보들을 의사결정 요소로서 고려한 의사결정의 동적 변환(Dynamic Adaptation)을 제안한다. 즉, 의사결정자의 정적, 외제적 정보보다 과거의 경험, 주관적 선호도 및 사회적 관계와 연관된 상황정보(Social Context)를 의사결정에 동적으로 반영하고 동시에 의사결정 해의 사용시점에서의 가용성에 따라 유용 가능한 대안을 추출하는 방법론을 제안하고자 한다. 이를 위해, 정적, 외제적 및 사회적 상황정보를 이용하여 의사결정 추론한다. 추론은 의사결정자의 과거 경험에 기반한 사례기반 추론과 해당 의사결정 결과가 가용하지 않을 경우 수정을 위한 제약식 만족추론으로 이루어진다. 이를 위해 개인적 경험 등의 정보에 기반한 '문제상황 온톨로지(Problem Context Ontology)와 집단의 경험적 지식에 기반한 '솔루션 온톨로지(Solution Ontology)를 구축하였다. 의사결정단계는 상황정보 인식 및 문제상황 온톨로지에 매핑하는 단계, 경험적 사례로부터 문제상황에 가장 적합한 사례를 선택하는 단계, 생성된 솔루션이 가용하지 않을 경우 솔루션 온톨로지와 제약식 만족추론을 통해 새로운 대안을 생성하는 단계로 이루어진다. 본 방법론을 도입에 적합한 식당을 제안하는 예제를 적용함으로써 타당성을 검증하였다. 또한 실험을 통해 사회적 상황정보를 고려하여 생성된 의사결정대안이 그렇지 않은 경우보다 의사결정자의 만족도를 향상시켰으며, 생성된 의사결정대안이 가용하지 않은 경우 제약조건식과 솔루션 온톨로지를 이용해 생성한 대안이 유의미함을 검증하였다.

.....
논문접수일 : 2011년 02월 15일 논문수정일 : 2011년 06월 24일 게재확정일 : 2011년 06월 27일

투고유형 : 국문일반 교신저자 : 이현정

1. 서론

유비쿼터스 컴퓨팅 환경은 동적으로 생성되는 정보의 인지와 활용을 가능하게 한다. 그러나 현재 인지되는 동적 정보에 비해, 이의 활용도는 낮은 편이다. 유비쿼터스 환경에서 의사결정을 해야 하는 개인 의사결정자들의 경우, 동적인 정보에 항상 노출되어 있으나 이를 실제 의사결정에 활용하는 데는 어려움이 있다. 이러한 어려움을 극복하기 위

해, 본 논문에서는 다양한 모바일 기기를 통해 인 지된 상황정보를 동적 의사결정자의 의사결정에 필요한 정보생성을 위한 동적변환을 지원하는 동적의사결정방법론을 제안한다. 이를 구현하기 위해서는 의사결정자가 필요로 하는 맞춤형 정보를 제공하는 것이 필요하며 이러한 맞춤형 정보를 제공하기 위해서는 동적 상황 정보를 정확히 인지하고 상황에 맞는 적절한 의사결정 해를 생성하는 것이 필요하다. 이를 위해 본 논문에서는 정보의 동

* 본 연구는 방위사업청과 국방과학연구소의 지원으로 수행되었습니다(UD110006MD).

적변환(Dynamic adaptation) 방법을 제안하였다. 즉 동적변환은 상황인지(context-aware)와 함께 유비쿼터스 컴퓨팅 환경의 핵심요소로서, 끊임없이 변화하는 상황정보에 노출되어 있는 의사결정자에게 시시각각 유용한 정보 제공을 위해 수행된다(Michele et al., 2010; Tayeb et al., 2004). 즉 의사결정자의 개인적 상황정보와 의사결정 대안에 귀속된 상황정보를 동시에 동적으로 모니터링 및 인지하여 이를 의사결정에 활용하여 가용한 최적의 의사결정을 이루도록 지원한다. 동적 변환을 위해서는 기 수립된 정책이나 업무관련 규칙 등을 활용해 규칙베이스를 구축해야 하며, 새롭게 인지된 상황 정보와 규칙베이스를 이용해 추론함으로써 맞춤형 정보를 생성하게 된다(Anders et al., 2003; John et al., 2003; Haake Jörg et al., 2009; Malandrino et al., 2010). 이때 중요한 것은 동일한 상황에 노출되더라도 의사결정자 개인의 선호도나 과거 경험 등에 따라 의사결정 결과가 달라질 수 있다는 것이다. 따라서, 특정 분야의 기 수립된 지식이나 업무관련 규칙 등을 일반화한 정보만을 이용해 맞춤형 정보를 생성하는 기존의 변환 방법으로는 양질의 의사결정을 기대할 수 없다. 또한 의사결정자 개인의 사회적 관계와 연관된 상황정보도 의사결정에 많은 영향을 미친다(Buriano, 2006). 그러나 정적 및 외부 환경적 상황정보에 비해 암묵지적인 성격이 강하기 때문에 식별과 모델링이 어렵고, 규칙형식에서의 일반화가 곤란하여 이를 규칙화 및 일반화하여 활용한 관련 연구는 거의 없다.

이에 본 연구에서는 의사결정자의 의사결정 시점에서의 정적, 외부적 환경의 상황정보뿐만 아니라, 의사결정자 개인의 사회적 상황정보를 고려한 의사결정시스템을 제안하였다. 이를 위해 상황정보를 표현하기 위한 지식표현 방법으로 온톨로지를 활용하였다. 온톨로지는 정보와 정보들 간의 관

계 표현이 용이하기 때문에 개인의 경험적 사회적 상황정보를 표현하기에 유용하다. 또한 온톨로지에 기반된 정보는 사례로 구축되었으며 개인의 상황정보에 따라 사례를 맞춤화하여 최적의 의사결정 해를 찾는 시스템을 제안한다. 즉, 사례기반 및 제약조건식 추론인 ‘상황 온톨로지를 이용한 동적 의사결정방법론’ 시스템은 다음과 같이 구성된다.

• 온톨로지 및 상황모델링

상황 모델링을 위한 온톨로지는 문제상황 온톨로지(Problem Context Ontology) 및 솔루션 온톨로지(Solution Ontology)로 구성된다. 문제상황 온톨로지는 의사결정자 개인의 모바일 기기에 위치하게 되며 솔루션 온톨로지는 서버에 위치한다. 문제상황 온톨로지는 개인의 의사결정을 지원한다. 솔루션 온톨로지는 집단지성에 기반해 다수 의사결정자의 의사결정 결과를 일반화 및 객관화 한다. 따라서 개인의 모바일에 저장되기보다 서비스 제공자의 서버에 위치하게 된다.

문제상황 온톨로지는 개인 의사결정자의 정적, 환경기반 동적 및 사회적 상황정보들의 외적, 내적 연관관계를 이용하여 의사결정결과를 맞춤화하도록 모델링 된다. 문헌에 의하면 상황정보 모델링을 위한 최적의 대안으로 온톨로지가 활용되고 있다(Thomas et al., 2004; Xioa et al., 2004; Ejigu et al., 2007). 따라서 본 연구에서도 상황정보 모델링을 위해 온톨로지를 사용하였다. 예를 들어 개인 ‘갑’이 회의를 위해 회의장소를 물색하는 경우 ‘갑’이 누구와 언제 무엇을 논의하느냐에 따라 즐겨 찾는 회의장소가 있을 수 있고 이러한 결과는 문제상황 온톨로지를 이용하여 사례형식으로 저장된다. 그러나 ‘갑’의 사례베이스에 저장된 회의장소가 의사결정자의 요구사항과 별개의 요인으로, 즉 그 회의장소가 가용하지 않다던가 등으로 인해

유사한 회의장소의 추천이 요구되는 경우 솔루션 온톨로지를 이용한다.

솔루션 온톨로지는 문제상황 온톨로지를 이용한 사례추론의 결과가 사용시점 가용하지 않고 문제상황 온톨로지로부터 또 다른 대안의 추출이 불가능한 경우 솔루션 온톨로지를 이용하여 대안을 찾는 것을 지원하도록 구축되었다. 즉, 솔루션 온톨로지는 일반적 의사결정자들의 상황정보를 이용한 의사결정 결과를 객관화 및 일반화하여 모델링 된다. 즉 '값'의 사례베이스에 저장되어 있지는 않지만 같은 상황조건, 누구와, 언제, 무엇을 회의할 것인가라는 조건하에 일반적으로 다른 사람들이 주로 이용하는 회의장소를 솔루션 온톨로지를 이용하여 사례베이스로부터 추출할 수 있다.

• 의사결정자의 과거 경험에 기반한 사례기반 추론

의사결정자가 상황정보에 노출되는 경우, 과거의 경험에 기반하여 의사결정을 추론하는 것이 일반적 의사결정 과정이다. 따라서 문제상황 온톨로지에 기반한 상황정보 모델링을 이용해 발생한 상황정보를 이용하여 개인의 과거의 유사 경험 사례베이스로부터 맞춤형된 사례를 추출한다.

• 상황변화에 따른 의사결정-개인-결과 수정을 위한 제약식 만족추론

의사결정자의 과거 경험을 이용한 사례기반추론으로 생성된 의사결정 대안이 현재 가용하지 않은 경우 유사 대안의 추출이 요구된다. 이를 위해 사례기반추론을 위해 사용된 상황정보와 추론의 결과로부터 온톨로지에 기반하여 제약식을 추출하며, 이들 제약식에 기반하여 일반적 사례베이스로부터 가능한 대안을 추출하기 위해 솔루션 온톨로지를 이용한 제약식 만족추론을 수행한다. 새로

이 탐색된 대안들 중 의사결정자의 요구사항을 만족시키는 해를 찾지 못하면 생성된 제약식 중의 사용자 요구사항의 우선순위에 따라 제약이 약한 조건식부터 조건 일부를 해제하면서 의사결정 해를 구한다.

본 연구는 상황정보 모델링과 사례기반 및 제약식 만족추론을 중심으로 상황정보기반 동적의사결정지원을 위한 시스템을 제안한다. 이를 위해 제 2장에서는 상황정보 및 상황모델링에 대해 문헌조사를 했으며, 제 3장에서는 '상황 온톨로지를 이용한 동적 의사결정시스템'의 구조를 제안하고 본 논문의 상황모델링을 제안한다. 제 4장에서는 추론을 위한 상황정보의 변환 및 문제상황 온톨로지를 이용한 매핑 및 사례기반 추론을 소개한다. 제 5장에서는 솔루션 온톨로지에 기반한 제약식 만족추론 방법론을 설명한다. 제 6장에서는 예제를 통해 본 시스템을 검증하고 제 7장에서는 실험을 통해 유용성을 확인하고 제 8장에서는 결론 및 추후 관련 연구에 대해 제안하였다.

2. 문헌 연구

의사결정자가 처한 상황은 다양하게 분류가 가능하다. 크게 의사결정자와 관련되어 쉽게 변하지 않는 정적정보, 외부환경요인에 따라 변하는 동적정보 및 의사결정자의 과거의 경험이나 사회적 관계를 고려한 사회적 정보로 나누어 볼 수 있다. 현재까지는 정적 및 환경적 상황정보에 기반하여 의사결정을 지원하였으나 의사결정자의 내면적 경험 등도 고려한 맞춤형된 의사결정의 요구가 증가하고 있다. 따라서 본 논문에서는 정적, 외부환경요인에 따라 변하는 동적정보와 개인의 내재적 상황요인으로 사회적 관계를 고려한 상황정보 및 상황정보모델링을 다음과 같이 문헌에 기반하여 살펴본다.

2.1 상황정보

본 논문에서는 상황정보를 최종 의사결정자들의 의사결정에 영향을 미치는 요인들로 정의하고 이들의 유형을 <표 1>과 같이 분류하면 다음과 같다.

<표 1> 상황정보 분류

상황정보유형		설명	예
정적 상황정보 (Static Context)		<ul style="list-style-type: none"> 이동통신장치에 저장된 사용자의 정보 환경변화에 따라 수정되지 않음 사용자 상황정보의 일부 	사용자 프로필, 옐로우 페이지
외부환경 상황정보 (Dynamic Context)	물리적 상황정보	<ul style="list-style-type: none"> 사용자 단말기 내재된 센서를 이용해 수집할 수 있는 상황정보 외부환경 상황정보 	위치, 시간
	환경적 상황정보	<ul style="list-style-type: none"> 외부 서비스 정보제공자로부터 수집할 수 있는 상황정보 물리적 상황정보의 일부 	날씨, 교통정보
사회적 상황정보 (Social Context)		<ul style="list-style-type: none"> 사용자와 상호작용을 하는 다른 사람들과의 관계 내부적 또는 논리적 상황정보의 일부 	사회적 관계, 상호관계의 목적, 흥미, 이해도 등

정적 상황정보로 정보사용자의 인구통계학적 정보인 나이, 성별 등을 들 수 있다. 이는 환경 변화에 따라 민감하게 반응하지 않는 특성이 있다. 또한 사용자 상황정보의 일부로 정의되기도 한다 (Guanling et al., 2000). 따라서 사용자의 이동통신 단말기에 저장이 가능하다. 예를 들어 사용자 프로필, 이동통신 단말기의 옐로우페이지 등에서 추

출할 수 있다. 이를 본 연구에서는 정적 상황정보 (Static context)로 분류한다. 이에 반해 외부환경의 상황정보는 내재된 센서를 이용해 인지하거나 정보 제공자로부터 제공받는다. 이동통신장치의 내재된 센서 등을 이용해 간단히 인지할 수 있는 상황정보로는 의사결정자의 위치, 시간 및 장소 등이 이에 포함되며 외부적 상황정보(external context) (Prekop et al., 2003; Gustavsen, 2002) 또는 물리적 상황정보(physical context) (Hofer et al., 2002)로 정의된다. 서비스를 제공하는 다른 정보 제공자로부터 제공받을 수 있는 외부환경정보로는 의사결정자를 둘러싸고 있는 교통, 날씨 등의 상황정보가 포함되며 외부환경적 상황정보(environmental context) (Daqing et al., 2004)로 분류된다. 본 연구에서는 외부환경요인에 따라 변하는 동적 정보로 분류하고 동적정보라 한다.

내재적 상황정보로는 의사결정자가 현재의 물리적 및 외부환경적 상황에서 상호작용을 하는 다른 의사결정자들과의 사회적 연관관계, 관심사의 공유도 및 친밀도 등을 포괄하는 정보를 고려할 수 있다. 이들 상황정보는 내적 상황정보(Prekop et al., 2003; Gustavsen, 2002), 논리적 상황정보 (Hofer et al., 2002), 사용자 상황정보(Daqing et al., 2004) 또는 사회적 상황정보(Anders et al., 2003; Guanling et al., 2000; Ghita et al., 2004; Ceri et al., 2007; Adams et al., 2008; Tran et al., 2009; Bin et al., 2004; Suganuma et al., 2008) 등으로 정의된다. 사회적 상황정보의 암묵지적 특성으로 인해 인지와 모델링이 어려워 연구에 한계가 있었으나, 본 연구에서 이를 모델링하여 의사결정에 반영하는 ‘상황 온톨로지를 이용한 동적의사결정방법론’을 제안하고 의사결정의 질을 향상시키고자 한다.

2.2 상황 모델링

스트랭과 린호프포피엔(Strang and LinnhoffPopien, 2004; Thomas et al., 2004)은 6개의 기준을 선정해 유비쿼터스 컴퓨팅 환경과 관련된 상황정보 모델링 방법에 대한 평가를 수행하였다. 평가 결과, 온톨로지 기반 모델링이 현재 환경에 실현가능성을 제외한 모든 측면에서 가장 우수한 것으로 입증되었다. 현재 환경에 실현가능성에서도 'Markup scheme modeling' 방법에 이어 두 번째로 좋은 방법론 중의 하나임을 제시한 바 있다. 따라서 수많은 연구들이 상황정보 모델링을 위해 온톨로지를 채택하고, 해결하고자 하는 문제의 특성에 적합한 상황정보 온톨로지를 구축해 활용하고 있다(Xiao et al., 2004).

이에 따라 본 논문에서는 동적 의사결정에 요구되는 상황정보를 사용자 상황정보인 정적, 외부환경기반 동적, 및 사회적 연관관계 정보로 구분하고 이들에 대해 다음과 같이 모델링한다.

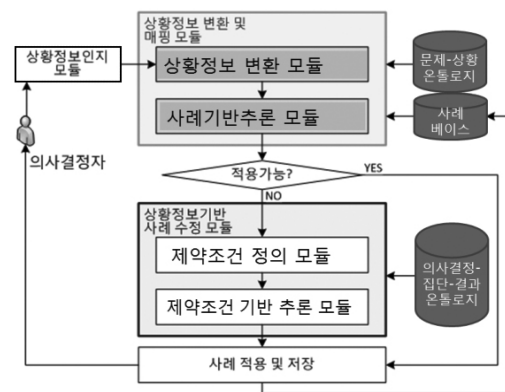
3. 상황정보 모델링

3.1 동적의사결정 추론 구조

상황정보를 이용한 동적의사결정방법론은 <그림 1>과 같이 상황정보 인지 모듈(Context-aware Module), 상황정보 변환 및 매칭 모듈(Context Conversion and Matching Module), 상황정보기반 사례수정 모듈(Context adaptation Module)과 사례 적용 및 저장 모듈(Case retain Module)로 구성된다. 본 연구에서는 '온톨로지를 이용한 동적의사결정방법론'의 추론과정인 상황정보-변환 및 매칭 모듈과 상황정보 기반 사례 수정 모듈에 중점 하도록 한다.

상황정보-변환 및 매칭 모듈은 인지된 상황정

보를 이용하여 문제상황 온톨로지를 이용하여 사례베이스로부터 의사결정 결과를 추론한다. 이 후 추론된 결과가 가용가능 하지 않은 경우 추출된 제약식을 이용하여 솔루션 온톨로지를 이용하여 대안을 추론하여 의사결정을 지원한다.

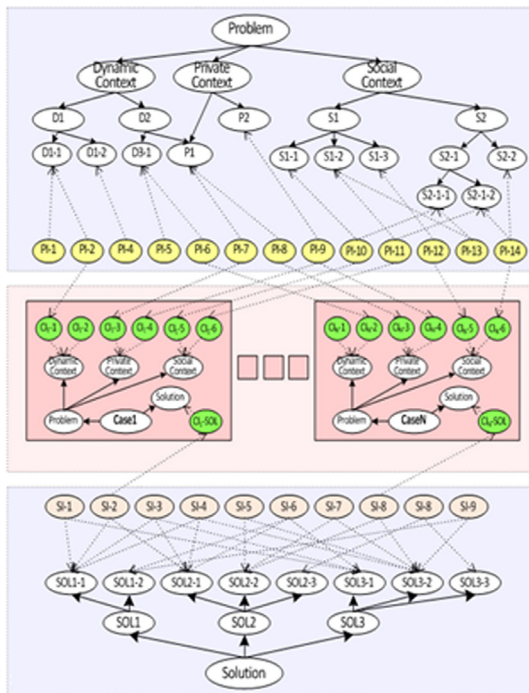


<그림 1> '상황 온톨로지를 이용한 동적의사결정방법론' 구조

3.2 온톨로지 구조

본 장에서는 사례기반추론 및 제약식 만족추론을 이용한 의사결정을 수행하기 위해 필요한 두 종류의 온톨로지를 제안한다. <그림 2>와 같이 제안하는 온톨로지 구조는 개인의 의사결정을 지원하는 문제상황 온톨로지(Problem Context Ontology)와 다수의 의사결정자의 의사결정 결과를 일반화 및 객관화한 솔루션 온톨로지(Solution Ontology)로 구성된다. 다음 그림에서 각 온톨로지는 정적, 환경기반 동적, 사회적 상황정보의 최상위 클래스와 하위 클래스와의 연관관계로 구성된다. 실선은 클래스간의 연관관계를 표현한다. 클래스와 점선으로 연결된 'P1-1' 또는 'S1-1' 등은 인스턴스를 나타낸다. 즉, 클래스와 인스턴스 및 인스턴스와 인스턴스의 관계는 점선으로 나타낸다.

• **문제상황 온톨로지(Problem Context Ontology; PO)** : 문제상황 온톨로지(PO)는 상황정보에 대해 상황의 유형을 결정하고 자신의 과거 의사결정 성향을 참조해 과거의 의사결정 결과들로부터 의사결정 해를 찾기 위해 사용된다. 이를 위해 발생 가능한 상황 정보들 간의 관계를 일반화하여 모델링 한다. 이 온톨로지의 최상위 클래스들은 세 가지 유형의 정적, 환경기반 동적, 사회적 상황정보에 연결된다. 그리고 각각의 클래스들과 개념적 연관 관계가 있는 상황정보를 수집해 하위 클래스들로 모델링 한다. 모든 상황정보의 단말 노드들에는 인스턴스들이 존재하며, 이 인스턴스들과 새로운 상황정보와 연결을 통해 사례의 문제상황을 생성한다. 문제상황 온톨로지의 구조는 <그림 2>에 도시되어 있다.



<그림 2> 문제상황 온톨로지, 솔루션 온톨로지 및 사례베이스

• **솔루션 온톨로지(Solution Ontology; SO)** : 솔루션 온톨로지(SO)는 문제상황 온톨로지(PO)를 이용해 선택된 대안이 가용하지 않은 경우에 일반적인 사례로부터 대안을 찾기 위해 사용된다. 즉, 솔루션 온톨로지는 일반 대중의 의사결정 결과들을 객관적인 관점에서 일반화 및 계층화 하여 구성된다. 따라서 여러 의사결정자들의 결과가 저장되어야 하고, 객관적인 관점에서 분류되어야 하므로 의사결정자 개인의 의사결정 과거 결과에 기반한 문제상황 온톨로지(PO)와는 독립적으로 모델링된다.

따라서 문제상황 온톨로지(PO)가 문제 해결을 위해 의사결정자 개인의 경험만으로 구축된 사례로부터 해를 찾는 것에 비해 솔루션 온톨로지(SO)는 유사한 경험을 가진 다수의 의사결정자들의 의사결정 결과들이 일반화된 사례들로부터 해를 찾도록 되어 있어 집단지성이 반영된 것이라고 할 수 있다.

3.3 상황정보 표현을 위한 사례 구조

본 연구에서는 온톨로지를 사례 정보 추론에 활용하였으며, 따라서 정보를 다음과 같이 사례구조화 하였다. 따라서 i^{th} 사례(CASE)의 상황정보를 식 (1)과 같이 표현하였다.

$$\begin{aligned}
 CASE^i &= \{P^i, S^i\} & (1) \\
 P^i &= \{(ct, cx_k^i, cv_k^i) \mid ct \in CT, cv_k^i \in POI, \\
 &\quad cx_k^i \text{는 문제상황의 인덱스}, k \geq 1\} \\
 CT &= \{1 : \text{정적}, 2 : \text{환경기반 동적}, 3 : \text{사회적}; \\
 &\quad \text{상황정보유형}\} \\
 POI &= \{cv_k \mid cv_k \text{는 PO의 인스턴스}\} \\
 PO &= \text{문제상황 온톨로지(problem Context Ontology)} \\
 S^i &= \{(sx_l^i, sv_l^i) \mid sx_l^i = \text{결론정보의 인덱스}, \\
 &\quad sv_l^i \in SOI, l \geq 1\} \\
 SOI &= \{sv_l^i \mid sv_l^i \text{는 SO의 인스턴스}\} \\
 SO &= \text{솔루션 온톨로지(Solution Ontology)}
 \end{aligned}$$

즉, i 번째 사례는 문제상황인 P^i 상황정보와 문제상황 온톨로지(PO)의 매핑으로 생성된다. S^i 는 문제상황에 따른 의사결정 솔루션이다. 사례기반 추론에 의해 생성되거나, 경우에 따라 제약식 만족 추론 과정에서 생성된다. 이와 같이 생성된 솔루션은 <그림 2>의 사례베이스에 사례의 형태로 저장된다.

4. 상황정보 변환 및 매핑모듈

4.1 상황정보 변환

본 연구에서 상황정보들은 인스턴스들의 집합 형식으로 표시된다. 따라서 획득된 상황정보들은 문제상황 온톨로지(PO)에 매핑시키는 작업이 필요하다. 예를 들어 본 논문의 예제인 레스토랑 선택 문제의 온톨로지의 인스턴스는 시간을 ‘식사시간대’와 ‘비식사시간대’로 나누고 있는 반면, 새로운 상황정보로 획득된 시간은 ‘13 : 30’과 같이 일반적인 시간일 경우가 이에 해당한다. 이럴 경우 일반 시간을 온톨로지에서 표현하고 있는 인스턴스와 해당 인스턴스의 상황정보 유형 값을 찾아 매핑한다.

이렇게 문제상황 온톨로지 매핑 시키기 위해 상황정보 값에 해당하는 문제상황 온톨로지의 인스턴스와 해당하는 상황정보 유형을 찾는다. 이는 상황정보유형(ct), 문제상황을 나타내는 인덱스(cx^{new}_k)와 인스턴스(cv^{new}_k)으로 구성된 순서쌍으로 표현한다. 위와 같은 변환과정을 거치고 나면 다음 식 (2)에서와 같이 문제상황 형식으로 상황정보가 모델링된다. 상황정보는 상황정보유형, 인덱스와 상황정보 값의 순서쌍의 집합으로 표현된다.

$$PO^{new} = \{(ct, cx^{new}_k, cv^{new}_k)\} \quad (2)$$

4.2 사례기반 추론

상황정보 변환에 의해 모델링된 상황정보를 이용하여 개인 의사결정자의 사례베이스를 기반으로 씨맨틱 유사도를 이용해 해를 탐색 추론한다. 씨맨틱 유사도는 상황정보와 사례와의 일치하는 또는 연관 있는 인스턴스의 수 및 상황정보 유형을 고려해 구한다. 유사도 식 (3)은 문제상황(Problem context)과 사례베이스의 사례로부터의 부분 온톨로지를 비교하게 된다. 부분 온톨로지의 노드는 클래스와 인스턴스로 구성되며, 이들은 연관관계(Arc)로 연결된다. 따라서 추출된 클래스와 인스턴스 및 연관관계를 포함하는 부분온톨로지를 비교한다. 이때 인스턴스는 클래스 및 관계를 포함해서 완전히 일치하는 인스턴스의 수와 클래스 공유 정도에 따른 관계에 따른 인스턴스의 수 등으로 가중치를 고려해 유사도를 계산한다. 유사도 값이 가장 큰 사례의 해가 의사결정자에게 제안된다.

$$sim(c_n, c_i) = \sum_i w_i \times \left(\frac{\sum_j w_{ij} \times n_{ij}}{\sum_j w_{ij}} \right) \quad (3)$$

n_{ij} 는 i^{th} 컨텍스트 유형 중 j^{th} 유형으로 매칭된 인스턴스의 수, $i = \{1 : 정적, 2 : 동적, 3 : 사회적\}$, $j = \{1 : exactly\ matched, 2 : subsumed\}$, w_{ij} 는 n_{ij} 의 가중치이고 w_i 는 컨텍스트 유형에 대한 가중치이다.

사례기반 추론에서 생성된 해를 의사결정자가 수용하면 식 (1)에서와 같은 형식의 사례가 생성되며, 생성된 사례는 사례베이스에 저장된다. 그러나 사례기반추론에 의해 제안된 대안이 의사결정자 개인의 상황정보(PO^{new})와는 관계없이 대안과 관련된 상황정보에 의해 사용시점에 유용하지 않으면, 상황정보 기반 사례 수정 모듈의 추론을 이용하여 새로운 대안을 추출한다.

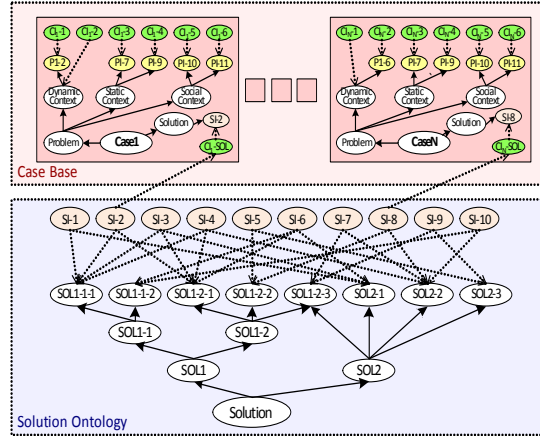
5. 상황정보 기반 사례 수정 모듈

5.1 제약조건 정의

의사결정자에게 사례기반 추론으로 제시된 해가 현재 상황에서 가용하지 않은 경우 다른 대안을 구하기 위해 솔루션 온톨로지와 제약식만족식 추론을 이용한다. 예를 들어, 사례기반추론에 의해 회의 장소로 제안된 회의장이 이미 예약되어 있어 사용이 불가능한 경우 새로운 회의장소의 탐색이 요구된다. 이럴 경우 본 논문에서는 사용자의 상황 특성에 기반해 선택된 대안의 특징을 표현하는 여러 제약식을 찾아내고, 이를 기반으로 다른 의사결정자들의 경험을 기반으로 구성된 전체 대안 집합인 솔루션 온톨로지(SO)를 이용하여 개인의 상황정보에 맞는 유사한 대안을 찾는다. 즉 솔루션 온톨로지(SO)는 제약식만족 문제를 효율적이면서 효과적으로 해결하기 위해 다수의 의사결정자의 결과를 객관적으로 분류한 온톨로지이다.

<그림 3>은 솔루션 온톨로지와 사례베이스에 저장된 사례와의 관계를 표현하고 있다. 즉, 하나의 사례와 솔루션 온톨로지(SO)와 연관관계는 사례기반 추론의 결과값과 솔루션 온톨로지(SO)의 인스턴스와 매핑으로 이루어진다. 예를 들어 Case 1의 결과값인 (CI_1-SOL) 은 솔루션 온톨로지(SO)의 인스턴스인 'SI-2'와 매핑되어 있다. 이 인스턴스는 솔루션 온톨로지의 두 개의 클래스 'SOL1-1-1', 'SOL1-2-1'에 연결된다.

사례는 솔루션 온톨로지와 연결되며, 사례와 연결된 솔루션 온톨로지 단말노드인 인스턴스의 해당 클래스를 추적하여 이를 특징짓는 제약식을 정의한다.



<그림 3> 솔루션 온톨로지와 연관된 사례베이스

5.2 제약조건 기반 추론

정의된 제약조건식은 의사결정 대안을 찾는 입력값이되면 이에 기반한 제약조건기반추론절차를 단계별로 살펴보면 다음과 같다.

Step 1 : 솔루션 온톨로지 탐색

사례기반추론을 통해 획득한 솔루션(S^{new})의 결과값(sv^{new})과 매핑되는 솔루션 온톨로지(SO)의 인스턴스를 찾아 인덱스(sx^{new})로 설정하여 (sx^{new} , sv^{new})로 매핑한다. 이를 위해서는 제 4장에서 제시한 상황정보와 문제상황 온톨로지(PO) 매핑방식과 같은 방법을 적용한다. 이 때 일치된 인스턴스가 존재하지 않는다면, 솔루션 온톨로지(SO)에 사례기반추론에서 찾아진 솔루션에 대한 정보가 존재하지 않으므로 솔루션 온톨로지(SO)에 본 사례의 솔루션(sv^{new})을 추가하는 작업을 선행 한다.

Step 2 : 대안 탐색

솔루션 온톨로지(SO)에서 특정 인스턴스에 연결된 클래스들을 그 인스턴스를 특징짓는 제약식으로 해석한다. 따라서 같은 클래스에 연결된 솔루션

선 온톨로지의 인스턴스들이 존재한다면 이들은 같은 제약식을 만족하는 대안으로 간주한다. 따라서 사례기반추론 결과에서 생성된 솔루션 sv^{new}_l 와 연결되어 있는 클래스의 집합 SC^{new} 을 식 (4)와 같이 구한다.

$$SC^{new} = (\{sc_i | sv^{new}_l \in ASSERT(sc_i), s_{ci} \in SO\}) \quad (4)$$

이렇게 SC 가 구해지면 모든 SC 에 연결되어 있는 인스턴스들의 교집합이 가능한 대안이 된다. 이때 현재 솔루션에 매핑되어 있는, 즉 대안으로서는 가능하나 현재 유용하지 않은 인스턴스 sv^{new}_l 를 제외한 대안들이 새로운 대안(AI)으로 선택된다.

$$AI = (\cap_{s_{ci} \in SC} ASSERT(s_{ci})) - sv^{new}_l \quad (5)$$

만약 새로운 대안(AI)의 개수가 0이면 ($|AI| = 0$) 대안을 찾기 위해 Step 4에서처럼 우선순위가 낮은 조건식부터 완화하여 대안을 찾는다. 이에 반해 대안이 두 개 이상이면 대안들 간의 우선순위를 정하기 위해 Step 3을 수행한다. 만약 하나이면 이를 대안으로 한다.

Step 3 : 제약 우선순위 결정

Step 2에서 도출된 2개 이상의 새로운 대안들 중에서 최적의 대안을 선정하는 것이 요구된다. 이때 선택된 모든 대안들이 공유하고 있는 제약식 이외에 각각의 대안들이 개별적으로 가지고 있는 추가적인 제약식들이 개인 의사결정자가 초기에 제시한 문제-상황과 얼마나 적합한 지를 평가함으로써 새로운 대안들 간에 우선순위를 결정한다. 이를 위해서는 추가적 제약식인 클래스에 연결되어 있는 인스턴스들을 구하고 해당 인스턴스들에 연결되어 있는 사례들의 적합도의 평균을 구한다. 이

때 적합도 평가는 사례기반추론에서 유사도 식 (3)에 의해 평가한다.

Step 4 : 제약 해제

Step 2로부터 대안이 도출되지 않은 경우, 사례기반추론 결과에서 생성된 솔루션 sv^{new}_l 가 가진 제약식들 중 우선순위가 낮은 제약식부터 해제하여 새로운 대안을 도출한다. 이 때 제약식 간의 우선순위는 연결되어 있는 인스턴스를 솔루션으로 갖는 사례들의 유사도 평균 값에 의해 결정된다.

이상의 온톨로지 기반의 제약식 만족추론에 의한 의사결정 대안선정을 위한 알고리즘은 다음 <그림 4>와 같다

```

main procedure MakeAlternatives

// 사례의 솔루션( $sv^{new}_l$ )과 솔루션온톨로지 매핑(Step 1)
RelatedSolutionInstance ← class of case solution instance
RelatedConstraints ← all classes which have 'instance-class' relationship with RelatedSolutionInstance

// 솔루션온톨로지로부터 대안탐색 (Step 2)
Alternatives ← GetAlternatives(RelatedConstraints)

// 대안이 2개 이상일 때 (Step 3)
if number of Alternatives >= 2 then
    Alternatives = SortAlternatives(Alternatives, RelatedConstraints)
end if

// 대안이 존재하지 않을 때 (Step 4)
if number of Alternatives = 0 then
    RelatedConstraints ← SortConstraints(Related Constraints)
    Alternatives ← RelaxConstraints(RelatedConstraints)
end if

// 대안이 1개 존재할 때
suggest the first Alternative
end procedure
    
```

```

// 우선순위가 낮은 제약식부터 해제
function RelaxConstraints(Constraints)
  repeat
    remove the first Constraint from Constraints
    Alternatives ← GetAlternatives(Constraints)
  until(number of Alternatives >= 1)

  Alternatives ← SortAlternatives(Alternatives, Constraints)
  return Alternatives
end function

// 대안추출
function GetAlternatives(Constraints)
  Alternatives ← all instances which have 'instance-class'
  relationship with the first Constraints
  for all Constraints
    CurrentAlternatives ← all instances which have
    'instance-class' relationship with current Constraints
    Alternatives ← Alternatives ∩ CurrentAlternatives
  end for
  return Alternatives
end function

// 대안들의 우선순위설정
function SortAlternatives(Alternatives, Constraints)
  ListofSimilarities ← NULL
  for all Alternatives
    CurrentConstraints ← all classes which have 'inst-
    ance-class' relationship with the current Alternative
    CurrentConstraints ← CurrentConstraints - Con-
    straints
    CurrentAlternatives ← GetAlternatives(Cur-
    rentConstraints)

    Similarity ← 0
    if number of CurrentAlternatives >= 1 then
      FilteredCases ← all cases of which solution is
      one of CurrentAlternatives

      if number of FilteredCases >= 1 then
        Similarity ← average of similarities between
        FilteredCases and current case
      end if
    end if
    add Similarity to end of ListofSimilarities
  end for

```

```

    sort Alternatives according to ListofSimilarities
    return Alternatives
  end function

// 대안들에 귀속된 제약식들의 우선순위설정
function SortConstraint(Constraints)
  ListofSimilarities ← NULL

  for all Constraints
    CurrentAlternatives ← GetAlternatives(Constraint)
    Similarity ← 0
    if number of CurrentAlternatives >= 1 then
      FilteredCases ← all cases of which solution is
      one of CurrentAlternatives
      if number of FilteredCases >= 1 then
        Similarity ← average of similarities between
        FilteredCases and current case
      end if
    end if
    add Similarity to end of ListofSimilarities
  end for

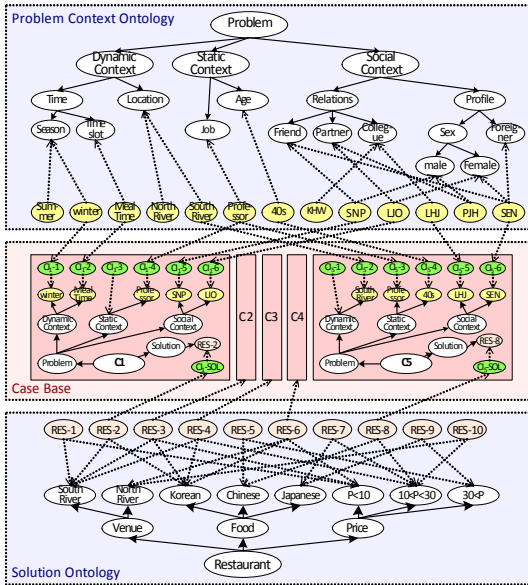
  sort Constraints according to ListofSimilarities
  return Constraints
end function

```

<그림 4> 솔루션 온톨로지-제약식 기반 알고리즘

6. 예제를 통한 방법론 검증

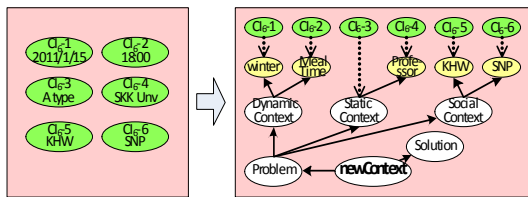
방법론의 논리성 검증하기 위해 모임의 성격에 따라 적합한 식당을 추천하는 예제를 살펴보기로 한다. 예를 위한 <그림 5>는 문제상황 온톨로지, 5개의 사례로 이루어진 사례베이스 및 솔루션 온톨로지로 이루어져 있다. 문제상황 온톨로지는 음식점 선정에 영향을 줄 수 있는 정적, 환경기반 동적, 사회적 상황정보로 이루어진다. 이 문제에서 솔루션 온톨로지에 해당하는 음식점들은 '음식 가격', '음식 종류', '음식점 위치'로 분류되었다.



<그림 5> 음식점 예제 온톨로지

6.1 상황정보 변환 및 유형 정의

‘상황 온톨로지를 이용한 동적의사결정방법론’은 상황정보 변환 모듈이 상황정보에 인덱스를 연결하고 상황유형을 결정한다. 다음 <그림 6>의 좌측 박스는 의사결정자에 귀속된 상황정보이며 문제상황 온톨로지(PO)를 이용해 다음과 같이 매핑된다.



<그림 6> 상황정보 변환 및 매핑

예를 들어 첫 번째 상황정보인 ‘CI6-1 : 2011/1/15’은 인덱스 ‘winter’에, 직장 ‘CI6-4 : SKK Univ’는 인덱스 ‘professor’에 그리고, ‘KHW’는 인덱스 ‘CI6-5 : KHW’에 연결된다. ‘winter’는 동적상황정

보에 ‘professor’는 정적상황정보, ‘KHW’는 사회적 상황정보의 유형으로 분류된다.

6.2 온톨로지를 이용한 사례기반 추론

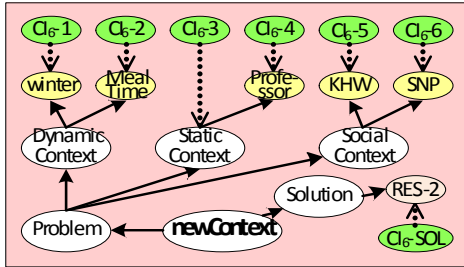
의사결정자의 사례베이스의 사례와 현재 입력된 상황정보와의 유사도를 식 (3)을 이용하여 평가한다. 제시된 유사도 평가를 위해 상황정보 유형간의 가중치 값을 사회적 정보는 2로 하고, 환경기반 동적 및 정적은 1로 가정한다. 비교 대상간의 일치성 평가를 위해 완전히 일치된 것만 가중치 1로 반영한다. 이럴 경우 <그림 7>의 새로운 상황 사례와 C1과 C5 각각의 대해 계산된 유사도는 다음과 같다.

$$sim(newContext, C1) = 1 \times (2) + 1 \times (1) + 2 \times (1) = 5$$

계산된 유사도 C1의 경우 환경기반 동적 유형의 클래스 가중치는 1이고, 이 두 개의 인스턴스 ‘winter’와 ‘Meal time’은 C1의 인스턴스와 모두 일치하므로 일치도는 2이다. 가중치 1의 정적상황정보 ‘professor’도 C1과 일치하므로 일치도는 1이다. 사회적 상황정보인 ‘KHW’는 가중치는 2이고, 2개의 인스턴스 중 1개의 인스턴스 ‘SNP’만 일치하므로 일치도는 1이다.

$$sim(newContext, C5) = 1 \times (0) + 1 \times (1) + 2 \times (0) = 1$$

계산된 유사도 C5의 경우 환경기반 동적 유형의 클래스 가중치는 1이고, 일치하는 인스턴스가 없으므로 일치도는 0이다. 가중치 1인 정적상황정보로 ‘professor’는 C1과 일치하므로 일치도는 1이다. 사회적 상황정보인 ‘KHW’는 가중치는 2이고, 2개의 인스턴스 중 어느 것도 일치하지 않으므로 일치도는 0이다.



<그림 7> 새로 생성된 상황 사례

이와 같이 C2, C3, C4에 대한 유사도는 $Sim(newContext, C2) = 2$, $Sim(newContext, C3) = 4$, $Sim(newContext, C4) = 2$ 로 계산된다.

따라서 C1이 의사결정자의 상황에 가장 유사한 결과 값으로 평가되어 이에 해당하는 음식점 'RES-2'가 해로 제안된다.

6.3 제약식만족 추론

사례기반 추론에서 선택된 해 'RES-2'가 현재 'RES-2'의 사정으로 예약이 불가능할 경우 다른 대안을 찾기 위해 다음의 프로세스를 이용한다.

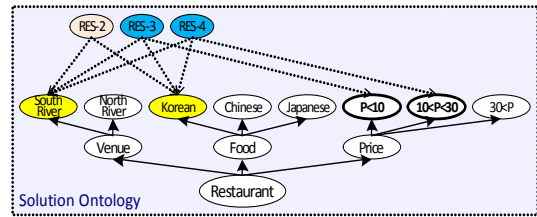
Step 1 : 솔루션 온톨로지 탐색

'RES-2'가 솔루션 온톨로지에 인스턴스로 존재하는 지를 확인 후, 존재하면 'RES-2'를 이용하여 제약식을 생성한다. 즉 'RES-2'에 연결되어 있는 클래스를 솔루션 온톨로지에서도 탐색한다. 탐색 결과 '음식점 위치'는 'South River'이고 '음식 종류'는 'Korean'임을 확인한다. 따라서 이 두 조건을 제약식으로 하여 다른 대안을 탐색한다.

Step 2 : 대안 탐색

솔루션 온톨로지 상에 클래스 <그림 8>의 'South River'에 연결된 대안들로 구성된 집합은 {RES-1, RES-2, RES-3, RES-4}이고 클래스 'Korean'을 만

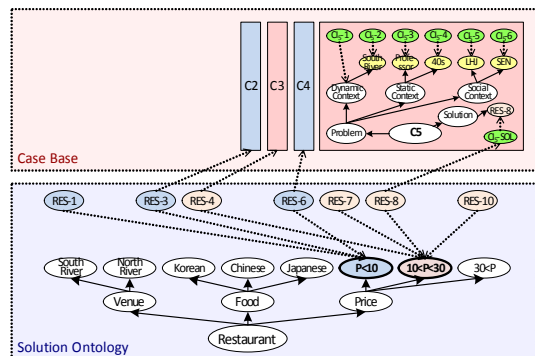
족하는 집합은 {RES-2, RES-3, RES-4, RES-6}이다. 따라서 현재 유용 가능하지 않은 'RES-2'를 제외하고 이들 클래스들 간의 교집합을 구하면 'RES-3'와 'RES-4'가 대안으로 가능하다.



<그림 8> 추가제약식 정의

Step 3. 대안 우선순위 결정

대안이 두 개 이상 존재할 경우 대안 들간의 우선순위는 식 (3)을 이용하여 평가한다. 즉, 이들 대안들을 포함하는 사례베이스의 사례들과 현재 사례 'C6' 간의 유사도를 평가하여 추가적으로 부여된 복수개의 제약식들 중 의사결정자의 현재 상황에 보다 적합한 제약식을 찾는다. <그림 6>에서 'RES-3'는 평균 가격대가 10\$ 미만의 식당이고 'RES-4'는 10\$에서 30\$ 사이의 레스토랑이다. 10\$ 미만의 레스토랑은 {RES-1, RES-3, RES-6}이고 10\$에서 30\$ 사이의 레스토랑은 {RES-4, RES-7, RES-8, RES-10}이다.



<그림 9> 추가제약식 기반 연관 사례 탐색

<그림 9>에서와 같이 사례베이스에서 10\$ 미만의 레스토랑들 중에 하나를 해로 갖는 사례는 {C2, C4}는 각각 인스턴스 {RES-3, RES-6}에 연결되며, 10\$에서 30\$ 사이의 사례 {C3, C5}는 인스턴스 {RES-4, RES-8}에 연결된다.

이들 사례베이스로부터 추출된 사례와 현재 상황을 고려하여 생성된 사례 'C6' 간의 평균 유사도를 앞서 계산했던 유사도 값을 이용하여 구하면 다음과 같다.

$$\begin{aligned} & \text{Average Similarity of } \{C2, C4\} \\ &= \{Sim(\text{newContext}, C2) + Sim(\text{newContext}, C4)\} / 2 \\ &= (2+2) / 2 = 2 \end{aligned}$$

$$\begin{aligned} & \text{Average Similarity of } \{C3, C5\} \\ &= \{Sim(\text{newContext}, C3) + Sim(\text{newContext}, C5)\} / 2 \\ &= (4+1) / 2 = 2.5 \end{aligned}$$

유사도 결과를 비교해 보면 제약식 '10 < P < 30'의 유사도가 2.5로써 제약식 'P < 10'의 2보다 높음을 알 수 있다. 따라서 제약식 '10 < P < 30'이 현재 문제와의 유사도를 높여주는 제약식으로 평가된다. 따라서 인스턴스들 중에 제약식 '10 < P < 30'을 만족하는 'RES-4'가 우선 추천 대상이 된다. 'RES-8'은 Step 2에 따라 고려대상에서 제외된다.

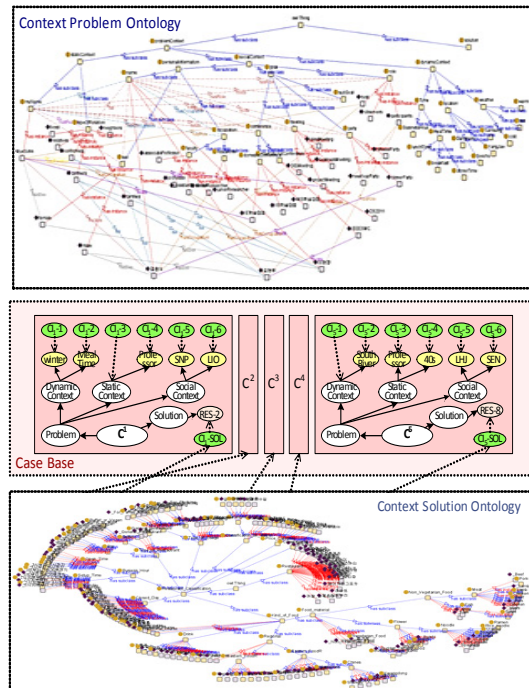
Step 4 : 제약식 해제

만약 사례기반추론에서 사례 C5가 선택되어 솔루션으로 'RES-8'이 추천되었다면 만족해야 하는 제약식은 'Chinese'와 '10 < P < 30'이다. 그러나 Step 2에 따르면 'RES-8'의 대안 제시가 불가능하다. 이럴 경우 'RES-8'이 가지는 두 제약식 'Chinese'와 '10 < P < 30' 중에 현재 문제와의 연관성이 낮은 제약식 순으로 조건을 해제하여 대안을 탐색한다.

대안 탐색은 Step 3의 프로세스를 따른다. 'RES-8'에서 중요도가 낮은 제약식 'Chinese'를 제거하면 대안으로 {RES-4, RES-7, RES-10} 선정되고 이들 복수 개의 대안은 식 (3)에 의해 유사도 평가 후 우선순위를 결정한다.

7. 실험 및 결과

본 논문에서는 실험을 위해 문제상황 온톨로지와 솔루션 온톨로지를 protégé를 이용하여 구축되었으며 윈도우 환경하에서 MS 비주얼베이직으로 다음과 <그림 10>과 같이 동적의사결정시스템의 프로토타입을 구축하였다.



<그림 10> 문제상황 온톨로지와 솔루션 온톨로지 프로토타입

본 실험을 위해 문제상황 온톨로지는 의사결정자

와 상호연관성을 가지는 친구, 동료, 가족들과의 연관관계 및 관계(예를 들어 만남 등)의 목적 등을 사회적 상황(social context)으로 모델링하였다. 이를 위해 사용된 클래스는 29개이며 인스턴스는 143개로 구성된다. 즉 상황정보는 문제상황 온톨로지를 활용해 모델링된다. 의사결정자의 상황정보는 문제상황 온톨로지에 포함된 인스턴스들의 부분집합으로 새로운 문제상황은 특정 부분집합에 포함된 인스턴스들이 갖는 모든 상위어(Hyperonym) 관계들로 구성된 부분 온톨로지가 된다. 선택된 해가 가용하지 않은 경우 제약조건식을 이용해 새로운 대안을 추출할 때 사용될 솔루션 온톨로지는 실제 식당 소개정보와 평가사이트의 평가 정보를 바탕으로 구성되었으며 29개의 클래스들과 76개의 인스턴스들로 구성되었다.

실험은 다음과 같이 이루어졌다. 첫째, 사회적 상황정보를 반영한 의사결정 대안과 사회적 상황정보를 반영하지 않은 대안 간의 의사결정자의 의사결정 만족도를 비교하여 그 차이가 있는 지를 비교한다. 둘째, 의사결정자 개인의 사회적 상황정보를 반영하여 선택된 의사결정 대안과 이 대안이 가용하지 않은 경우에 이에 대한 대안으로 제약조건식에 의해 새롭게 선택된 대안과의 의사결정 만족도 비교를 통해 두 그룹간의 의사결정자의 만족도 차이가 있는지를 비교하였다. 의사결정자의 의사결정을 위해 시스템에 의해 제시된 대안에 대한 만족도 측정은 두 실험 모두 5점 척도를 이용하였다. 매우 만족은 5점으로 표시하도록 하였으며 이에 따른 각 가설에 따른 실험은 다음과 같다.

첫 번째 실험은 사회적 상황정보를 고려해 의사결정 대안을 도출한 경우와 사회적 상황정보 고려 없이 대안을 도출한 경우에 대해 각각의 그룹에 대해 실험은 105회 실시되었다. 실험에 의해 도출된 대안에 대한 의사결정자의 만족도 비교를 위해

쌍별비교 t-test를 이용하였다. 그 결과 두 그룹간의 차이의 유의미성($t = 2.626, p < 0.005$)을 확인하였다. 즉 사회적 상황정보를 고려한 그룹의 만족도가 고려하지 않은 그룹의 만족도보다 높음을 확인하였다.

두 번째 실험에서는 사회적 상황정보를 고려해서 문제상황 온톨로지로부터 생성된 의사결정 대안이 현재 상황에서 가용하지 않은 경우 제약조건식의 완화를 이용해 솔루션 온톨로지로부터 새로운 대안을 생성하였다. 즉 사회적 상황정보의 고려에 의해 생성된 대안과 제약조건식 완화를 통해 생성된 두 대안들 간의 의사결정자의 만족도를 비교하였다. 이를 위해 실험은 90회 실시되었으며 두 집단의 만족도 차이 비교를 위해 쌍별비교 t-test를 적용하였으며 두 그룹간의 차이가 미약함($t = 1.19, p < 0.15$)을 확인하였다. 비록 제약식을 완화했다더라도, 솔루션 온톨로지에서 선택된 해와 개인의 정보를 바탕으로 생성된 문제상황 온톨로지에 기반해 선택된 해 사이의 만족도의 차이는 적었다. 즉 현재의 가용하지 않은 해의 대안으로 제약식 완화를 통해 새로운 대안을 선택하여 의사결정자에게 제시한 결과 의사결정자는 새로운 대안을 수용할 수 있음을 확인하였다.

유비쿼터스 환경에서 의사결정자를 둘러싼 상황의 유동성으로 인해 의사결정자의 의사결정 결과의 가용성이 시시각각으로 장소, 방법, 시간 등에 의해 변화하는 경우 이에 대한 가용한 의사결정 대안의 제시가 요구된다. 본 연구에서는 사용자의 사회경험적 상황정보에 기반해 생성된 문제상황 온톨로지로부터 선택된 해가 사용시점에 가용하지 않은 경우 제약식 완화 알고리즘을 통해 집단 지성에 기반한 솔루션 온톨로지로부터 의사결정자에게 유의미한 대안을 제시해 줄 수 있음을 확인하였다.

8. 결론

본 논문에서는 상황을 정적정보, 환경기반 동적 정보, 사회 경험적 정보로 분류하여 '상황 온톨로지를 이용한 동적 의사결정방법론'을 제안하였다.

본 방법론은 다음과 같이 구성된다. 의사결정자의 의사결정에 영향을 주는 정적, 환경기반 동적, 사회적 상황정보를 효과적으로 표현하기 위한 문제상황 온톨로지와 사례기반 추론을 통한 의사결정자의 개인적 사회 경험적 사례에 기반한 의사결정 대안의 추출, 및 개인적 경험의 한계를 극복하기 위한 방법으로 다수의 의사결정자들의 사회적 경험을 객관적 관점으로 체계화한 솔루션 온톨로지의 제시이다. 또한 문제상황 온톨로지로부터 생성된 대안이 가용하지 않은 경우 제약조건 완화를 통해 솔루션 온톨로지로부터 의사결정 대안을 새롭게 생성할 수 있도록 하였다.

본 논문은 기존의 온톨로지 구축에서 고려하지 않았던 사회적 상황정보의 고려가 의사결정자의 의사결정 만족도의 향상에 미치는 영향과 개인의 사회적 경험에 의해 선택된 대안이 다수의 의사결정자의 경험에 의해 생성된 대안과의 차이점이 있는지를 확인하였다. 실험을 통해 사회적 상황정보의 고려가 의사결정자의 만족도를 향상시킬 수 있는 대안의 생성을 가능하게 함을 확인하였고, 개인의 경험에 기반해 제시된 대안이 가용하지 않은 경우 차선으로 솔루션 온톨로지로부터 제약식 완화를 통해 추출된 대안이 유의미함을 확인하였다.

참고문헌

Adams, B., P. Dinh and V. Svetha, "Sensing and Using Social Context", *ACM Transactions on Multimedia Computing, Communications and Applications*, Vol.5, No.2(2008), 11~37.

Anders, K. and A. Agnar, "Case-Based Situation Assessment in a Mobile Context-Aware System", *Proceedings of AIMS2003, Workshop on Artificial Intelligence for Mobil Systems*, Seattle, October, 2003.

Bin, W., B. John and S. K. S. Gupta, "Supporting Persistent Social Groups in Ubiquitous Computing Environments Using Context-Aware Ephemeral Group Service", *Proceedings of the Second IEEE Annual Conference on Pervasive Computing and Communications (PERCOM'04)*, 2004.

Buriano, L., "Exploiting Social Context Information in Context-Aware Mobile Tourism Guides", *In Proceeding of Mobile Guide, 2006*, Turin, Italy, 2006.

Guanling, C. and K. David, "A Survey of Context-Aware Mobile Computing Research", *Dartmouth Computer Science Technical Report TR2000*, 2000.

Ceri, S., F. Daniel, M. Marera and F. M. Facca, "Model-Driven Development of Context-Aware Web Applications", *ACM Transactions on Internet Technology*, Vol.7, No.1 (2007), 1~33.

Daqing, Z., Y. Zhiwen and C. Chung-Yau, "Context-Aware Infrastructure for Personalized Healthcare", *The International Workshop on Personalized Health*, December, 13~15, Belfast, Northern Ireland, 2004.

Ejigu, D., S. Marian and B. Lionel, "Semantic Approach to Context Management and Reasoning in Ubiquitous Context-Aware Systems", *IEEE*, 2007.

Ghita, M. G., P. Jacques and B. Patrick, "Context-Aware Computing : A Guide for the Pervasive Computing Community", *Proceedings of the IEEE/ACS International Conference on Pervasive Services (ICPS'04)*, 2004.

- Gustavsen, M. R., "Condor—an application framework for mobility-based context-aware applications", *Proceedings of the Workshop on Concepts and Models for Ubiquitous Computing*, Goeteborg, Sweden, 2002.
- Haake Jörg, H., H. Tim, B. Joop, L. Stephan, V. Dirk and Z. Ziegler, "Context Modeling for Adaptive Collaboration", *Technische Berichte der Abteilung für Informatik und Angewandte Kognitionswissenschaft*, 2009.
- Hofer, T., W. Schwinger, M. Pichler, G. Leonhartberger and J. Altmann, "Context-awareness on mobile devices—the hydrogen approach", *Proceedings of the 36th Annual Hawaii International Conference on System Sciences*, 2002, 292~302.
- John, K. and C. Vinny, "A Policy-Driven, Context-Aware, Dynamic Adaptation", *Fourth IEEE International Workshop on Policies for Distributed Systems and Networks (POLICY03)*, 2003.
- Malandrino, D., M. Francesca, R. Daniele, B. Claudio, C. Michele and S. Vittorio, "MIMOSA : context-aware adaptation for ubiquitous web access", *Personal and Ubiquitous Computing*, Vol.14, No.4(2010), 301~320.
- Michele, S., E. Sebastian, R. Franco, R. S. David and W. Zhimin, "Context-Aware Adaptive Applications: Fault Patterns and Their Automated Identification", *IEEE Transactions on Software Engineering*, Vol.36, No.5(2010), 644~661.
- Prekop, P. and M. Burnett, "Activities, context and ubiquitous computing", *Special Issue on Ubiquitous Computing Computer Communications*, Vol.26, No.11(2003), 1168~1176.
- Suganuma, T., Y. Kazuhiro, T. Yoshikazu, T. Hideyuki and S. Norio, "A Ubiquitous Supervisory System based on Social Context Awareness", *The 22nd International Conference on Advanced Information Networking and Applications*, 2008.
- Tayeb, L. and L. Nabil, "Context-Aware Adaptation for Mobile Devices", *In Proceedings of IEEE International Conference on Mobile Data Management*, 2004, 106~111.
- Thomas, C. and L. Claudia, "A Context Modeling Survey", *Workshop on Advanced Context Modelling Reasoning and Management as part of UbiCom*, 2004, 33~40.
- Tran, M. H., J. Han and A. Colman, "Social Context : Supporting Interaction Awareness in Ubiquitous Environments", *In Proceedings of the 6th Annual International Conference on Mobile and Ubiquitous Systems (MobiQuitous'09)*, 10, Toronto, Canada, July 2009.
- Xiao, W. H. and Z. D. Qing, "Ontology Based Context Modeling and Reasoning using OWL", *Second IEEE Annual Conference on Pervasive Computing and Communications Workshops*, Orlando, Florida, March, 14~17, 2004.

Abstract

Dynamic Decision Making using Social Context based on Ontology

Hyunwoo Kim* · Mye Sohn** · Hyun Jung Lee***

In this research, we propose a dynamic decision making using social context based on ontology. Dynamic adaptation is adopted for the high qualified decision making, which is defined as creation of proper information using contexts depending on decision maker's state of affairs in ubiquitous computing environment. Thereby, the context for the dynamic adaptation is classified as a static, dynamic and social context. Static context contains personal explicit information like demographic data. Dynamic context like weather or traffic information is provided by external information service provider. Finally, social context implies much more implicit knowledge such as social relationship than the other two-type context, but it is not easy to extract any implied tacit knowledge as well as generalized rules from the information. So, it was not easy for the social context to apply into dynamic adaptation. In this light, we tried the social context into the dynamic adaptation to generate context-appropriate personalized information. It is necessary to build modeling methodology to adopt dynamic adaptation using the context. The proposed context modeling used ontology and cases which are best to represent tacit and unstructured knowledge such as social context.

Case-based reasoning and constraint satisfaction problem is applied into the dynamic decision making system for the dynamic adaption. Case-based reasoning is used case to represent the context including social, dynamic and static and to extract personalized knowledge from the personalized case-base. Constraint satisfaction problem is used when the selected case through the case-based reasoning needs dynamic adaptation, since it is usual to adapt the selected case because context can be changed timely according to environment status.

The case-base reasoning adopts problem context for effective representation of static, dynamic and social context, which use a case structure with index and solution and problem ontology of decision maker. The case is stored in case-base as a repository of a decision maker's personal experience and knowledge.

The constraint satisfaction problem use solution ontology which is extracted from collective

* Graduate School of EEWS, KAIST

** Department of Systems Management Engineering, Sungkyunkwan University

*** Graduate School of Business, Sogang University

intelligence which is generalized from solutions of decision makers. The solution ontology is retrieved to find proper solution depending on the decision maker's context when it is necessary. At the same time, dynamic adaptation is applied to adapt the selected case using solution ontology.

The decision making process is comprised of following steps. First, whenever the system aware new context, the system converses the context into problem context ontology with case structure. Any context is defined by a case with a formal knowledge representation structure. Thereby, social context as implicit knowledge is also represented a formal form like a case. In addition, for the context modeling, ontology is also adopted. Second, we select a proper case as a decision making solution from decision maker's personal case-base. We convince that the selected case should be the best case depending on context related to decision maker's current status as well as decision maker's requirements. However, it is possible to change the environment and context around the decision maker and it is necessary to adapt the selected case. Third, if the selected case is not available or the decision maker doesn't satisfy according to the newly arrived context, then constraint satisfaction problem and solution ontology is applied to derive new solution for the decision maker. The constraint satisfaction problem uses to the previously selected case to adopt and solution ontology. The verification of the proposed methodology is processed by searching a meeting place according to the decision maker's requirements and context, the extracted solution shows the satisfaction depending on meeting purpose.

Key Words : Social Context, Ontology, Case-Based Reasoning, Constraint Satisfaction Problem

저자 소개



김현우

현재 KAIST EEWS 대학원 초빙교수로 재직 중이다. KAIST에서 산업공학을 전공하고, 경영정보학 석사와 경영공학 박사학위를 취득하였다. 주요 관심분야는 녹색 기술 사업화 전략, 지능형정보시스템, 사례기반추론 등이다.



손미애

성균관대학교 산업공학과를 졸업하고 한국과학기술원에서 경영정보공학으로 박사 학위를 취득한 후, 한국국방연구원에 근무했으며, 현재 성균관대학교 시스템경영공학과 부교수로 재직 중이다. 주요 관심분야는 시멘틱 웹, 시멘틱 웹 서비스, BPMS 및 Enterprise Architecture/Architecture Framework 등이며, 현재 온톨로지상에서의 유사도 계산과 SBA를 위한 EA 구축연구를 수행하고 있다. 한국 경영공학회와 ITA/EA 학회의 임원으로 활동 중이며, CACM을 비롯한 국내외 우수 저널에 논문을 발표하고 있다.



이현정

이화여자대학교 물리학과 졸업 및 전자계산학 석사를 받았으며, 한국과학기술원(KAIST) 테크노경영대학원 경영공학 박사학위를 취득하였다. 현재 서강대학교 BK21 연구교수로 재직 중이다. 주요 연구분야는 지능정보시스템, 의사결정지원시스템, 유비쿼터스 컴퓨팅 및 시멘틱 웹 서비스 등이다.