

온톨로지 기반 상황해석구조를 이용한 의도추론의 모호성 해결[☆]

Solving the ambiguity of an Intention Reasoning using Context-Awareness Architecture based on Ontology

이 승 철* 김 치 수** 임 재 현***
Seung-Chul Lee Chi-Su Kim Jae-Hyun Lim

요 약

온톨로지를 이용한 상황인식 시스템은 추론엔진의 도움을 받아 상황을 추론할 수 있다. 추론엔진의 도움을 받고, 추론규칙 문법에 맞는 추론규칙을 작성함으로써 기존 상황인식 시스템이 가진 추론의 모호성을 해결할 수 있다. 또한 추론 알고리즘을 프로그램으로부터 배제함으로써 새로운 상황에 보다 쉽게 적용할 수 있는 장점을 가진다. 본 논문에서는 온톨로지를 이용한 상황인식 시스템을 제안한다. 또한 온톨로지를 이용한 상황인식 시스템의 효용성을 확인하기 위해 가정을 대상으로 한 구현과 실험을 실시하였다.

Abstract

Context-Aware system using ontology is able to infer a context from help by reasoning engine. It can solve the ambiguity of intention reasoning of context-aware system as it is being made a reasoning rule followed reasoning grammar and being helped by reasoning engine. Also, it has a merit that is easy to apply to new environment by excluding reasoning algorithm from the program. In this paper, we are present context-aware system using ontology. We have tested and implemented it at home basis environment to verify of its effectiveness.

☞ keywords : context-awareness, ubiquitous, ontology, intention reasoning, 상황인식, 유비쿼터스, 온톨로지, 의도추론

1. 서 론

유비쿼터스 컴퓨팅 환경에서 가장 핵심적인 기술은 상황인식 기술이다[1]. 상황인식 기술을 이용하는 상황인식 서비스는 사용자의 의도와 상황을 이해하기 위해 많은 데이터를 필요로 한다. 이러한 데이터는 각종 센서들에 의해 입력되기도 하고, 내부 메모리 혹은 데이터베이스에 저장되어

있기도 하다. 상황인식 서비스를 위한 상황해석 구조는 이러한 데이터를 처리하고, 데이터를 이용해 상황을 추론하기 위한 프로그램 혹은 미들웨어의 중심적인 구조로 사용된다. 프로그램 내에 상황해석 구조와 데이터 구조를 가지고 있는 상황인식 시스템은 상황의 변화에 적절히 대응하기 힘든 단점을 가지고 있다[2].

이를 개선하기 위해 온톨로지를 이용하고자 하는 연구가 지속적으로 진행 중이다. 대표적인 연구로는 메릴랜드 대학의 Harry Chen의 주도로 이루어진 CoBra Ontology[3][4], 상황인식 시스템에서의 표준적인 온톨로지 적용을 위한 SOUPA[5], 다양한 모바일기기에서 이용하기 위한 Celadon Ontology[6], 이벤트에 따라 실시간으로 구조를 변

* 준 회원 : 공주대학교 대학원 컴퓨터공학과 박사과정
scin21c@kongju.ac.kr

** 정 회원 : 공주대학교 컴퓨터공학부 교수
cskim@kongju.ac.kr

*** 종신회원 : 공주대학교 컴퓨터공학부 교수
defacto@kongju.ac.kr

[2007/05/30 투고 - 2007/06/18 심사 - 2007/08/06 심사완료]

☆ 본 연구는 공주대학교 자체학술연구지원으로 수행되었음

경하는 SOA[7]등이 있다. 이러한 연구들은 특정한 지역, 특정한 서비스를 대상으로 하고 있다.

온톨로지는 데이터 사이의 관계를 생성하고, 컴퓨터가 데이터를 이해할 수 있도록 도와준다 [2]. 상황을 해석하기 위해서는 수많은 데이터가 필요하며 데이터는 인간에 의해 입력되거나, 센서를 통해 입력된다. 하지만 데이터는 인간이 이해하기 쉬운 문자로 구성되어 있어 컴퓨터가 단어의 의미를 이해한다는 것은 불가능하다. Gruber는 온톨로지를 “공유된 개념화(shared conceptualization)에 대한 정형화되고 명시적인 명세(formal and explicit specification)”라고 정의하였다[3]. 온톨로지의 특징 중 하나인 규칙(rule)을 이용한 추론을 함으로써 스스로 관계를 정의하고, 필요한 정보를 추론할 수 있다[4]. 온톨로지를 이용한 상황인식 시스템에서는 온톨로지 구조와 규칙을 재 정의하고 배포함으로써 변화된 상황 혹은 전혀 다른 상황에서도 상황인식 시스템의 프로그램 수정 없이 적용이 가능하다[5]. 온톨로지를 이용한 상황인식 시스템은 프로그램 위주의 상황인식 시스템에 비해 관리, 유지, 보수에 편의성이 높다고 볼 수 있다.

본 논문에서는 유비쿼터스 환경을 위한 상황해석 구조를 제안한다. 기존의 상황인식 시스템은 새로운 상황에 대응하기가 쉽지 않다. 기존의 상황인식 시스템은 상황해석을 프로그램에서 수행하고 있다. 새로운 상황에 적용하기 위해서는 프로그램의 수정이 필요하고, 상황이 복잡해짐에 따라 프로그램 또한 복잡해진다. 이렇게 프로그램의 복잡도가 상승하게 되면, 프로그램은 상황에 더욱 대응하기 힘들어진다. 또한 상황을 추론함에 있어 하나의 상황을 두 가지 이상으로 해석하거나 다른 상황에 대해 동일하게 해석하는 추론의 충돌이 발생할 수도 있다. 추론의 충돌은 상황의 추론을 어렵게 한다. 이러한 상황해석의 문제를 온톨로지를 이용하여 줄일 수 있다. 온톨로지는 구조화된 포맷을 가진다. 구조화된 포맷을 가진 온톨로지는 추론엔진의 지원을 받을 수 있다. 추론엔진은 정해진 문법에 따라 작성된 규칙을 이용

하여 추론을 한다. 온톨로지를 이용한 추론엔진을 이용함으로써 추론의 충돌 등으로 인한 상황해석의 모호성을 줄일 수 있다.

본 논문에서는 온톨로지를 이용하여 프로그램의 복잡성을 낮추고, 해석의 모호성을 보완할 수 있는 온톨로지를 이용한 상황해석 구조를 제안한다. 온톨로지를 이용한 상황해석 구조의 실험을 위해 가정 내에서의 안전 서비스를 제공해 주는 온톨로지 구조와 규칙을 작성하고 실제 프로그램을 개발하여 실험한다. 또한 해석구조의 검증과 성능 평가를 위해 온톨로지를 사용하지 않은 상황해석 구조와 환경 변화 시 수정사항을 비교하고, 온톨로지의 기술적 평가를 위해 Protégé 2000을 이용한 ontology validation[14]을 수행하였다. 이와 함께 온톨로지의 완성도를 평가하기 위해 다중 사용자 환경에서의 동일한 상황에서의 두 가지 상황해석이 이루지는 시나리오를 통해 추론의 모호성이 발생할 수 있는 환경에서 추론이 명확히 이루어지는가를 실험하였다.

2. Ontology 기반의 상황해석구조

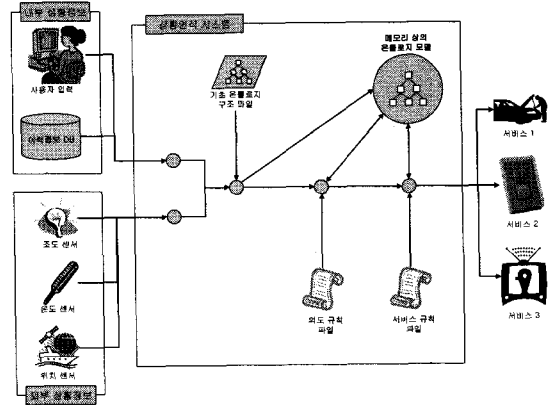
2.1. 온톨로지를 이용한 상황해석 프로세스

상황해석을 위해서는 여러 가지 데이터가 필요하다. 이러한 데이터는 상황인식 서비스가 필요한 상황에 따라 달라진다. 또한 데이터는 센서로부터 입력되거나, 사용자에 의해 입력될 수 있으며, 기존의 데이터를 사용할 수도 있다. 데이터의 구분과 데이터의 성격에 따라 5가지의 요소를 정의하였다[15]. 이 5가지의 요소는 데이터를 수집하고 상황을 해석하기 위해 필요한 요소로써, “Object”, “Time”, “Location”, “Action” 그리고 “Intention”이다 [6]. Object, Time, Location, Action은 입력된 데이터를 위한 상황요소로써 최종적으로 Intention을 파악하기 위한 기초 데이터가 된다.

표 1은 상황해석 구조의 5가지 상황요소를 설명한다.

(표 1) 상황해석 구조의 5가지 상황요소

	정의	등록 가능한 데이터
Object	대상 정보. 서비스 받는 주체 혹은 서비스를 행하는 주체.	- 대상의 이름, 성별, 나이 등 - 대상의 분류 (사람, 자동차, 컴퓨터 등)
Time	시간 정보 날짜, 시간과 관련된 어떠한 정보	- 현재 날짜, 시간 - 계절
Location	위치 정보. Object의 위치와 관련된 정보. 실제 위치 좌표일 수도 있으며, 추상적인 위치.	- 위치 좌표 - 대상이 위치한 장소명 - 장소의 추상적 위치 (건물, 차량, 지하, 지상 등)
Action	행동 정보. Object가 행한 행동과 관련된 정보	- 실제 행위 (앉다, 일어서다, 들어오다, 나가다 등) - 행위와 관련된 정보 (누운 후 경과한 시간 등)
Intention	현재의 상황 정보. 상황 판단에 따른 정보로써 현재의 상황 혹은 대상이 요구하는 서비스 상황.	

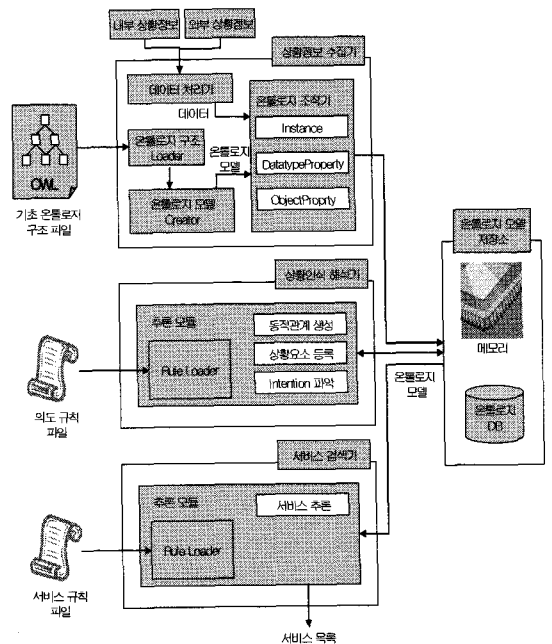


(그림 1) 상황해석 과정

1차 추론을 통해 온톨로지 모델이 데이터, 의도 규칙 등을 통해 완전한 하나의 구조로 만들어진다. 이렇게 만들어진 구조는 서비스 규칙을 통한 2차 추론을 통해 서비스 검색에 이용된다.

그림 2는 온톨로지를 이용한 상황해석 구조의 상세 프로세스를 보여준다.

온톨로지를 이용한 상황해석은 그림 1과 같은 과정을 가진다. 상황인식 시스템은 기초 온톨로지 구조를 읽어 들이고 입력되는 데이터를 처리하여 기초 온톨로지 구조를 입력한다. 데이터가 입력됨으로써 단순히 클래스와 프로퍼티의 구조를 가지고 있던 기초 온톨로지 구조는 하위 클래스와 인스턴스들, 그리고 데이터 프로퍼티의 실제 값이 채워진다. 데이터와 프로퍼티 관계가 형성된 기초 온톨로지 구조는 온톨로지 모델로 불린다. 이렇게 데이터가 입력된 온톨로지 모델은 1차 추론을 거쳐 완전한 온톨로지 모델로 형성된다. 1차 추론에서는 각 데이터(클래스, 인스턴스) 간의 하위 관계가 생성되고, 의도 규칙에 의한 데이터 간의 관계가 생성된다. 또한 1차 추론을 거침으로써 상황 해석에 필요한 5가지 요소의 값이 생성되고, Intention이 파악된다. 1차 추론을 거친 온톨로지 모델은 서비스 규칙을 이용한 2차 추론을 거쳐 서비스 검색이 이루어진다.



(그림 2) 온톨로지를 이용한 상황해석 구조의 상세 프로세스

2.2. 상황정보 수집기

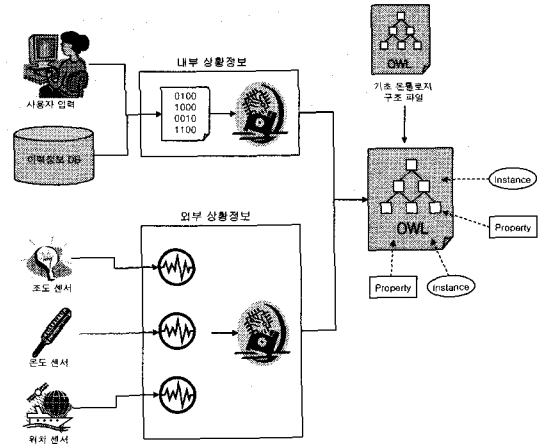
상황정보 수집기는 데이터를 처리하고 온톨로지 모델에 입력하는 역할을 한다. 상황정보 수집기는 내부 상황정보와 외부 상황정보를 수집한다. 내부 상황정보는 데이터베이스, 파일, 메모리 등에 저장되어 있는 데이터 혹은 상황에 따라 사용자가 입력하게 되는 입력 데이터 등이 있다. 이 정보는 미리 입력된 사용자의 신원정보, 서비스되었던 이력정보, 사용자 취향정보 등으로 기존에 입력되어 있어야 하거나, 누적되어야 할 데이터들이다. 외부 상황정보는 센서 등에 의해 입력되는 데이터이다. 상황인식에 필요한 모든 센서가 대상이 될 수 있다. 이러한 센서들은 사용자를 인식하거나 사용자의 위치를 추적하고, 사용자의 행위를 인지할 수 있다. 그림 3은 상황정보 수집기의 전체 프로세스를 보여준다.

기초 온톨로지 구조가 메모리에 등록된 후 상황정보 수집기는 데이터를 수집한다. 센서링 등을 통해 이벤트 시점이 되면, 입력된 데이터는 온톨로지 모델에 입력된다. 우선 내·외부 상황정보들이 온톨로지 모델에 정의된 각 클래스들의 인스턴스나 DatatypeProperty로 생성된다. DatatypeProperty는 이러한 인스턴스들 혹은 클래스들이 가지게 되는 실제 값으로써 사용자의 이름, 나이 등 대상의 특징들이다. 이들이 이후 관계의 선언, 추론 등에 사용되는 실제 데이터들이다. 데이터가 입력된 후 인스턴스와 데이터, 인스턴스와 클래스, 인스턴스들 간의 관계가 ObjectProperty를 통해 연결된다.

2.3. 상황인식 해석기

상황인식 해석기의 최종적인 목표는 사용자의 의도(Intention)의 파악이다. 이를 위해 상황인식 해석기는 온톨로지 구조에 의도 규칙(Intention Rule)을 적용한다.

상황인식 해석기는 작성된 의도 규칙을 읽어



(그림 3) 상황정보 수집기의 프로세스

들이다. 의도 규칙에는 세 종류의 규칙이 담겨 있다. 첫 번째 규칙은 상황에 따른 관계의 생성으로 추론에 의해 생성될 동적인 관계를 담고 있다. 두 번째 규칙은 Object, Time, Location, Action 요소의 값 생성이다. 값들은 온톨로지 구조에 포함된 데이터들에 기반 하여 생성된다. 생성된 값들은 최종적으로 Intention을 파악하는데 사용된다. 마지막 세 번째 규칙은 Intention 해석을 위한 규칙이다. 두 번째 규칙을 통해 각 요소에 생성된 값들을 통해 Intention이 해석된다.

상황인식 해석기는 메모리에서 온톨로지 모델을 불러오고, 파일로 저장된 의도 규칙 파일을 읽어 들인다. 의도 규칙 파일의 첫 번째 규칙에 따라 각 대상들의 동적인 관계를 설정한다. 동적인 관계는 상황에 따라 관계가 바뀌어야 하는 것들이다. 동적인 관계가 생성된 후 Object, Time, Location, Action의 요소 값이 생성된다. 이 네 가지의 요소에 입력되는 데이터는 상황인식 서비스가 필요한 곳에 따라 달라질 수 있다. 요소의 값은 단어로 입력된다. 기존의 상황인식 시스템이 데이터 검색을 위해 특정 코드 값을 사용했던 것과 달리 온톨로지를 사용한 상황해석 시스템에서는 관계를 통해 데이터를 연결하므로 컴퓨터 중심의 코드 값이 아닌 프로그래머, 관리자 등 인간 중심의 단어를 사용할 수 있다. Object는 서비스

를 받는 대상이 될 수 있지만, 서비스를 실행하는 주체가 될 수도 있다. 또한 Object가 서비스를 받는 대상으로 설정된다 하더라도 대상의 이름을 사용할 것인지, 성별을 사용할 것인지 등을 선택해야 한다. 본 논문에서 제안하는 온톨로지를 이용한 상황해석 구조에서는 상황의 해석에 키워드를 사용한다. Object, Time, Location, Action의 상황요소에 사용할 수 있는 키워드를 정의해 둔 후 규칙을 통해 적합한 키워드와 연결한다.

Intention은 위의 네 가지 요소의 값에 따라 판단된다. Intention 파악을 위한 규칙은 네 가지 요소의 키워드를 통해 지금의 상황이 어떠한 상황인지 판단한다. 물론 온톨로지 모델에 상황에 따른 데이터와의 관계가 설정이 되어 있어야 한다. Intention의 파악에 필요한 키워드는 규칙에 정의된다. 규칙에 정의되는 키워드는 해당 Intention에 필요한 모든 키워드가 아닌 중요한 키워드만이 정의되어 있다.

2.4. 서비스 검색기와 서비스 실행기

서비스 검색기는 Intention에 의해 판단된 상황에 따른 적절한 서비스를 검색한다. 서비스들은 온톨로지 모델에 등록되어 있다. 각각의 서비스는 특정한 그룹을 가지고 있지 않으며 개별적으로 등록된다.

Intention이 판단된 후 서비스 검색기는 서비스 규칙 파일을 읽어 들인다. 서비스 규칙 파일은 실행 가능한 모든 서비스의 적용 범위가 작성되어 있으며, 판단된 Intention에 적합한 서비스들이 검색된다. 온톨로지 구조에서 각각의 서비스들은 적용 가능한 Intention과 executeIntention의 ObjectProperty와 관계를 가지고 있다. 표 2는 서비스를 추출하는 규칙이다. 규칙에 정의된 바와 같이 추론된 Intention과 executeIntention으로 연결된 서비스는 모두 현재 실행되어야 하는 서비스로 추론된다.

서비스 실행기는 서비스 검색기에 의해 검색된

(표 2) 서비스 검색 규칙

```
[rule_service: (?X pre:hasIntention ?Y),(?service pre:executeIntention ?Y) -> (pre:Intention pre:hasService ?service)]
```

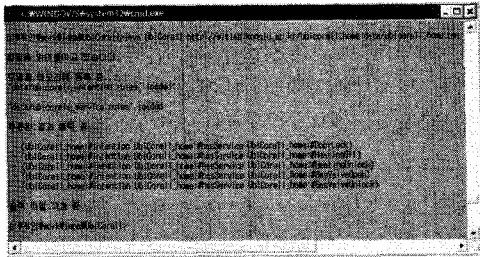
서비스를 실행하는 역할을 한다. 누적되어 저장되어야 할 이력정보가 필요한 경우 이력정보를 저장하는 것도 서비스 실행기의 역할이다. 사용자의 취향을 파악하기 위해서는 사용자에게 실행되었던 서비스 정보를 누적시켜 분석하는 것이 효과적이다. 이를 위해 서비스 실행기는 현재의 Intention과 실행된 서비스의 목록을 저장한다.

3. 실험

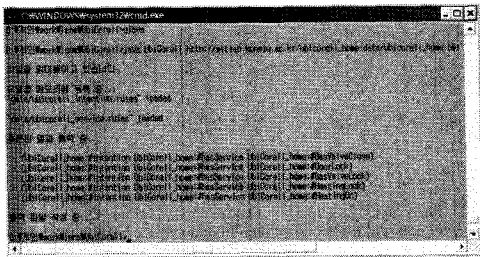
본 논문에서는 온톨로지를 이용한 상황해석 구조를 제안한다. 상황해석 구조에 온톨로지가 유용하게 사용될 수 있음을 증명하기 위해 실제 기초 온톨로지 구조를 작성하고 의도 규칙과 서비스 규칙을 작성하여 실험하였다[10]. 기초 온톨로지 구조의 작성은 OWL DL을 준수하는 구조로 작성하였다[12]. 상황해석 처리를 위한 프로그램은 Java2 SE 1.5.2를 기반으로 작성하였으며, 온톨로지의 처리 및 추론을 위해 HP Lab에서 개발한 Jena2 Framework를 사용하였다[13].

3.1. 시나리오

실험의 시나리오는 가정을 대상으로 한다. 가정 내에서 구성원에 따라 적합한 상황인식 서비스를 실행해 줄 수 있는가가 실험의 목표이다. 서비스를 받는 대상은 가정의 구성원으로 성인과 어린아이이다. 가정 내의 현재 구성원이 누구인가에 따라 각기 다른 서비스가 실행된다. 구성원에서의 어린아이는 가정 내의 장치에 대한 작동이 서투르거나 오작동을 할 가능성이 높은 10세 미만의 유아로 제한한다. 또한 외출을 2가지로 구분하여 단기외출과 장기외출로 정의하였다. 단기의



<SecureKeep Intention>



<SecureAll Intention>

(그림 5) 추론된 실험 결과

3.3. 실험 결과 고찰

3.3.1. 비교평가

온톨로지를 이용한 상황해석 구조와 온톨로지를 사용하지 않은 상황해석 구조는 앞서 언급한 다양한 상황을 적용하기 위한 프로그램의 수정의 복잡성으로 비교할 수 있다. 이러한 비교를 위해 상황의 변화에 대한 각각의 수정 사항을 비교하였다.

(표 3) 프로그램 수정 비교

온톨로지 이용	온톨로지를 이용하지 않음
<ul style="list-style-type: none"> • 사용자 클래스에 인스턴스 추가 • 대상에 따른 키워드 추가 • 새로운 상황에 대한 Intention 해석 규칙 추가 	<ul style="list-style-type: none"> • 새로운 사용자에게 대한 새로운 변수 선언(혹은 변수 타입 변경) • DB의 사용자 테이블 구조 변경(필요 시) • 새로운 사용자에게 대한 코드 값 추가 • 대상 판단 알고리즘 추가 • 상황해석 알고리즘 수정(새로운 대상에 대한 Intention 추가)

표 3은 새로운 서비스 대상이 추가됨에 따라 온톨로지를 이용한 경우와 이용하지 않은 경우의 프로그램 수정 사항의 간략한 비교이다.

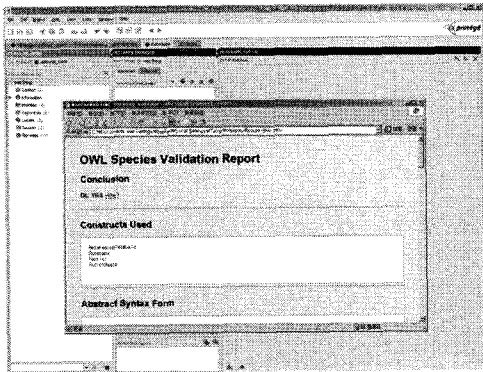
반면 온톨로지를 이용한 상황해석 구조를 사용한 시스템에서는 컴파일 된 프로그램의 수정 없이 온톨로지 구조와 규칙의 변경만으로 적절한 대응이 가능하다. 온톨로지는 클래스와 인스턴스 구조를 가진다. 따라서 사용자와 관련된 클래스를 정의해 두었다면 다른 사용자가 추가되더라도 구조의 변화 없이 사용자 인스턴스가 동적으로 추가됨으로써 시스템 스스로 능동적인 대응이 가능하다.

3.3.2. 온톨로지 평가

2가지 이상의 상황이 충돌하여 발생할 수 있는 상황해석의 모호성은 시스템이 얼마나 데이터의 특성을 정확히 파악하고 데이터의 관계를 설정하느냐에 따라 해결될 수 있다. 보다 정확한 상황의 파악을 위해서는 더 많은 데이터가 필요하다. 하지만 데이터가 많아질수록 온톨로지의 구조는 복잡해지고, 결과적으로 명확하지 못한 온톨로지 구조가 생성될 수 있다. 이러한 온톨로지 구조는 결국 추론 상에서의 오류와 두 가지 이상의 상황에서 상황 추론을 결정짓지 못하는 추론의 모호성을 가질 수 있다.

따라서 온톨로지의 구조상에 오류가 없는지, 온톨로지를 구현한 구문에 문제는 없는지를 검사하는 것은 온톨로지를 이용한 상황 추론에서 중요한 평가가 될 수 있다. 우선 온톨로지 개발 툴을 이용하여 온톨로지의 Validation을 수행할 수 있는데, 이 Validation은 구문 검사, 순환성, 일관성 등을 검사할 수 있다[9]. 온톨로지의 기술적 평가를 위해 온톨로지 개발 툴인 Protégé 2000의 plug-in을 이용하여 온톨로지 구조의 Validation을 수행하였다. 그리고 온톨로지의 완성도를 평가하기 위해 실험의 시나리오를 이용하여 추론 결과를 확인하였다.

본 논문에서 제안하는 온톨로지 구조는 OWL DL을 준수하여 작성하였으므로 명확한 추론이 가능하며, 이를 온톨로지 개발 툴의 Validation을 통해 검증하였다. 아래 그림 6은 Protégé 2000 plug-in을 이용한 온톨로지 Validation 결과이다.



(그림 6) 온톨로지 Validation 결과

온톨로지의 완성도 평가를 위해 실험의 시나리오를 통해 적절한 서비스가 추론되는지 실험하였다.

실험에서는 다중 사용자를 고려한 온톨로지 구조를 변경 없이 사용하였으며, 추론 규칙 또한 변경 없이 사용하였다. 실험은 서비스 대상이 어른 1명인 환경과 어른 1인, 어린이 1인인 환경, 어른 2인인 환경을 대상으로 하였다. 실험의 목적은 서비스 대상이 다른 환경에서 적절한 서비스가 추론되는가이다. 어른 1명만이 존재하는 환경에서 어린이를 대상으로 하는 서비스가 추론되지 않는지, 혹은 추론에 오류가 발생하지 않는지 실험하였다.

(표 4) 각 환경에서의 서비스 추론 결과

환경	조건 (사용자, 상태, 상태변화 경과시간)	의도 추론 결과
어른	어른, 외출, 00:19:00	SecureKeep
	어른, 외출, 00:32:00	SecureAll
	어른, 귀가, 00:13:00	SecureNot

어른, 어린이	어른, 외출, 00:32:00	SecureAll
	어린이, 외출, 00:45:00	
	어른, 외출, 00:12:00	SecureKeep
	어린이, 외출, 00:25:00	
	어른, 외출, 00:54:00	SecureKids
	어린이, 귀가, 00:11:00	
	어른, 귀가, 00:05:00	SecureNot
	어린이, 귀가, 00:35:00	
어른1, 어른2	어른, 귀가, 00:47:00	SecureNot
	어린이, 외출, 00:12:00	
	어른1, 외출, 00:19:00	SecureKeep
	어른2, 외출, 00:45:00	
	어른1, 외출, 00:32:00	SecureAll
	어른2, 외출, 00:58:00	
	어른1, 귀가, 00:02:00	SecureNot
	어른2, 외출, 01:08:00	
어른1, 어른2	어른1, 귀가, 00:13:00	SecureNot
	어른2, 귀가, 00:01:00	

위의 표 4는 각 환경에서의 추론 결과표이다. 표를 통해서 각기 다른 환경에서도 정상적인 추론이 발생함을 알 수 있다. 서비스 대상이 1명에서 2명으로 바뀌더라도 서비스 대상에 따른 의도가 추론된다. 또한 서비스 대상이 2명인 환경에서도 대상에 따라 의도가 추론된다. 어른이 2명인 환경에서 어린이에 대한 의도인 SecureKids가 추론되지 않는 것을 볼 수 있다.

이는 규칙을 통해 사용자가 어른인지 어린이인지 파악하고, 그에 따라 추론을 하기 때문이다. 이에 따라 어린이를 대상으로 하는 의도인 SecureKids는 어른이 1명인 환경과 어른이 2명인 환경에서 해당 대상인 어린이가 없기 때문에 추론되지 않는다.

실험을 통해 본 논문에서 제안하는 온톨로지 구조가 서비스 대상이 바뀌더라도 일관성 있는 추론을 할 수 있음을 알 수 있다.

4. 결론

본 논문에서는 온톨로지를 이용한 상황해석 구조를 제안하였다. 상황인식 시스템은 다양한 환경

에서 이용될 수 있다. 이를 위해 상황인식 시스템의 상황해석 구조는 다양한 환경에 쉽게 적용할 수 있어야 한다. 하지만 기존의 상황인식 시스템은 프로그램 구조를 이용해 상황을 해석함으로써 새로운 상황에 쉽게 대응하기 어렵다. 또한 복잡한 상황에서 프로그램 구조가 복잡해짐에 따라 상황해석에 오류가 발생할 가능성도 높아진다.

온톨로지를 이용한 상황해석 구조는 프로그램의 복잡도를 낮춰 프로그램의 수정을 최소화함으로써 새로운 상황에 쉽게 대응할 수 있다. 또한 온톨로지 추론엔진을 이용함으로써 정해진 문법 내에서 규칙을 작성한다. 추론엔진을 이용함으로써 프로그램 구조에 기반을 둔 상황해석보다 상황해석의 모호성을 줄일 수 있다. 실제 가정을 대상으로 하는 상황인식 시스템을 개발하고 실험을 실시하였다. 또한 실험결과를 분석하여 상황해석 구조에 온톨로지를 이용함으로써 프로그램의 복잡도를 낮추고, 추론을 통한 상황해석의 모호성이 어느 정도 줄일 수 있음을 확인하였다.

하지만 해석해야 할 상황이 늘어나고, 수집되는 정보가 늘어날수록 온톨로지의 구조와 규칙은 더욱 복잡해진다. 그에 따라 온톨로지 구조의 설계와 규칙의 설계에 많은 시간이 소요된다. 그리고 정확한 상황의 해석을 위해서는 온톨로지 구조와 규칙의 철저한 검증과 여러 차례의 테스트가 필요하다. 온톨로지 구조가 복잡해짐에 따라 온톨로지 모델의 생성에 소요되는 시간이 증가하고, 온톨로지 모델의 조작에 보다 많은 시간이 소요될 것이다. 보다 정확한 상황의 인식을 위해서 규칙이 복잡해짐에 따라 추론시간도 증가할 것이다.

온톨로지를 이용한 상황해석 구조를 상황인식 서비스에 사용하기 위해서는 높은 수준의 온톨로지 설계와 규칙 설계가 필요하다. 하지만 온톨로지 구축을 위한 GUI기반 도구들의 지속적인 발전과 온톨로지 추론 엔진의 발전은 상황해석에 온톨로지를 보다 쉽게 적용할 수 있도록 도와줄 것이다. 본 연구는 이러한 온톨로지 기술의 발전

과 함께 정확한 상황해석 구조로 발전해 나갈 것이다.

참고 문헌

- [1] S. S. Yau and F. Karim, "An Adaptive Middleware for Context-Sensitive Communications for Real-Time Applications in Ubiquitous Computing Environments", *Real-Time Systems*, 26(1):29-61, January 2004.
- [2] Christopoulou, E., Goumopoulos, C., Zaharakis, I., Kameas, A., "An Ontology-based Conceptual Model for Composing Context-Aware Applications", *Research Academic Computer Technology Institute* (2004).
- [3] Chen, H. and Finin, T., "An Ontology for Context-Aware Pervasive Computing Environments", *The Knowledge Engineering Review archive* Volume 18, Issue 3, pp. 197-207, 2003.
- [4] H. Chen, T. Finin, A. Joshi, "An Intelligent Broker for Context-Aware Systems", *InCollection, Adjunct Proceeding of Ubicomp 2003*, October 2003.
- [5] Chen, H., Perich, F., Finin, T., and Joshi A., "SOUPA: Standard Ontology for Ubiquitous and Pervasive Applications". In *Proceedings of the First Annual International Conference on Mobile and Ubiquitous systems: Networking and Services (Mobiquitous 2004)*, Boston, MA, 2004.
- [6] Lee, M.C., Jang, H.K., Paik, Y.S., Jin, S.E. and Lee, S., "Ubiquitous Device Collaboration Infrastructure: Celadon", *The Fourth IEEE Workshop on Software Technologies for Future Embedded and Ubiquitous Systems, and the Second International Workshop on Collaborative Computing, Integration, and Assurance*, 2006.
- [7] Cheng, S.Y., Jin, W.R. and Hsu, Y.J.,

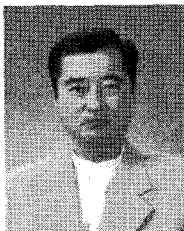
- "Context-aware Policy Matching in Event-Driven Architecture", Technical report, AAAI Press, Menlo Park, California, pp. 140-141, 2005.
- [8] T. Berners-Lee, J. Hendler, and O. Lassila, "The sementic web", Scientific American, May 2001.
- [9] Gruber, T., "A Translation Approach to Portable Ontologies", Knowledge Acquisition, Vol. 5, No. 2, pp.199-220, 1993.
- [10] A. Harth and S. Decker. "OWL Lite-Reasoning with Rules", WSML Working Draft, 2005.
- [11] Harry Chen, Tim Finin Anupam Joshi, "A Context Broker for Building Smart Meeting Rooms", in the Proceeding of the Knowledge Representation and Ontology for Autonomous Systems Symposium, 2004 AAAI Spring Symposium, Stanford University, CA, March 2004.
- [12] <http://www.w3.org/TR/owl-features/>
- [13] <http://jena.sourceforge.net/>
- [14] A. Gomez-Perez, and M.C. Suarez-Figueroa, "Results of Taxonomic Evaluation of RDF(S) and DAML+OIL ontologies using RDF(S) and DAML+OIL Validation Tools and Ontology Platforms import services", EON 2003 2nd International Workshop on Evaluation of Ontology-based Tools, 2003.
- [15] 임재현, "지능형 서비스를 위한 상황인식 해석 구조 설계 및 구현", 한국인터넷정보학회 논문지, 제7권5호, 2006. 10.

● 저 자 소 개 ●



이 승 철(Seung-Chul Lee)

2005년 공주대학교 정보과학전공 졸업(학사)
 2007년 공주대학교 대학원 멀티미디어공학과 졸업(석사)
 현재 공주대학교 대학원 컴퓨터공학과 박사과정
 관심분야 : 상황인식, RFID, 온톨로지
 E-mail : scin21c@kongju.ac.kr



김 치 수(Chi-Su Kim)

1984년 중앙대학교 전자계산학과 졸업(학사)
 1986년 중앙대학교 대학원 전자계산학과 졸업(석사)
 1990년 중앙대학교 대학원 전자계산학과 졸업(박사)
 1992~현재 공주대학교 컴퓨터공학부 교수
 관심분야 : 컴포넌트 개발 방법론, 온톨로지 개발 방법론 etc.
 E-mail : cskim@kongju.ac.kr



임 재 현(Jae-Hyun Lim)

1986년 중앙대학교 전자계산학과 졸업(학사)
 1988년 중앙대학교 대학원 전자계산학과 졸업(석사)
 1998년 중앙대학교 대학원 컴퓨터공학과 졸업(박사)
 1998~현재 공주대학교 컴퓨터공학부 교수
 관심분야 : 상황인식, RFID/USN, 온톨로지, 인터넷 기술
 E-mail : defacto@kongju.ac.kr