

논문 2008-45CI-5-19

계층적 분류체계를 지원하는 규칙기반 추론엔진

(A Rule-based Reasoning Engine supporting Hierarchical Taxonomy)

김태현*, 김재호*, 원광호*, 이기혁**, 손기락***

(Taehyun Kim, Jaeho Kim, Kwangho Won, Kihyuk Lee, and Kirack Sohn)

요약

미래 유비쿼터스 컴퓨팅은 언제 어디서나 지능형 모바일 단말들이 자율적으로 서비스를 제공받을 수 있는 유비쿼터스 지능 공간을 필요로 한다^{[1][2]}. 이러한 지능 공간의 자율적 구성을 위해 지능 공간에 속한 각 모바일 단말들은 다양한 소스로부터 컨텍스트(Context, 상황) 정보를 수집하고 컨텍스트 정보로부터 유용한 정보를 추론할 수 있어야 한다. 특히 다양한 유비쿼터스 지능 공간으로부터 수집되는 컨텍스트 정보의 모호성을 극복하고 보다 정확한 상황 인지를 통한 지능형 서비스를 제공하기 위해서는 컨텍스트에 대한 표준 분류 기법(taxonomy) 및 분류된 컨텍스트 정보를 기반으로 하는 추론 기술이 요구된다. 이를 위해 기존의 유비쿼터스 지능 공간에 관련된 대부분의 기존의 연구들^{[1][2][3]}에서는 상황·인지 서비스 제공을 위해 CLIPS^[4]나 JESS^[5]와 같은 규칙 기반 추론 엔진이 주로 사용되고 있다. 그러나 기존의 추론 엔진들은 리소스가 제한된 모바일 단말에서 사용되기에는 한계를 가지고 있다. 따라서 본 논문에서는 모바일 단말을 위한 자율적인 상황인지 서비스를 제공하기 위한 경량 추론 엔진을 설계하고 구현하는 것을 목적으로 한다. 개발된 추론 엔진은 휴대폰이나 PMP, 네비게이션 등과 같은 개인형 모바일 단말에서 자율적인 상황인지 기반 서비스를 제공하기 위해 사용될 수 있다. 또한, 계층적 분류체계(taxonomy) 정보를 활용함으로써 일반적인 룰(general rule) 또는 구체적인 룰(specific rule)의 선택적인 구성을 통해 다양한 수준의 컨텍스트가 실시간으로 수집되는 상황인지 컴퓨팅에서의 효율적인 상황인지 서비스의 구현을 지원한다.

Abstract

In a ubiquitous computing environment, a ubiquitous smart space is required to help devices provide intelligent services^{[1][2]}. The smart space embedded with mobile devices should have the capabilities of collecting data and refining the data to context. Unfortunately, the context information in a ubiquitous smart space has many ambiguous characteristics. Therefore, it is necessary to adapt a standard taxonomy for context information in the smart space and to implement an inference technique of the context information based on taxonomy. Rule-based inference engine, such as CLIPS^[4], Jess^[5], was employed for providing situation-aware services^{[1][2][3]}. However, it is difficult for those engines to be used in resource limited mobile devices. In this paper, we propose a light-weight inference engine providing autonomous situation aware services in mobile environment. It can be utilized for personal mobile devices such as mobile phone, PMP and navigation. It can also support both generalized rules and specialized rules as using hierarchical taxonomy information.

Keywords : Taxonomy, Rule-based Reasoning, Situation Awareness, Rete

* 정희원, 전자부품연구원 유비쿼터스연구센터

(Ubiquitous Technology Research Center, Korea Electronics Technology Institute)

** 학생회원, *** 정희원, 한국외국어대학교 컴퓨터및정보통신공학부

(School of Computer Science and Information Communications Engineering, Hankuk University of Foreign Studies)

※ 본 연구는 21세기 프론티어 연구개발사업의 일환으로 추진되고 있는 지식경제부의 유비쿼터스컴퓨팅 및 네트워크원천기술개발사업의 08B3-01-10M 과제로 지원된 것임

접수일자: 2008년8월20일, 수정완료일: 2008년9월8일

I. 서론

인간의 컴퓨팅 환경은 기존의 단순한 PC 중심의 데이터 프로세싱 개념에서 네트워킹 기반의 유비쿼터스 컴퓨팅 개념으로 발전해 가면서 개인형 모바일 플랫폼 기술에 대한 중요도가 대두되고 있다. 특히, 유비쿼터스 컴퓨팅에서의 개인형 모바일 플랫폼 기술은 재구성형 네트워킹(reconfiguration networking) 기술을 중심으로 발전하여왔으나 최근의 연구는 상황인지(situation awareness) 기술의 융합에 관심이 모아지고 있다.

모바일 환경에서의 상황인지 서비스는 소형 모바일 단말이 가지는 리소스의 한계뿐만 아니라 높은 이동성과 같은 특수성을 갖는다. 이로 인해 제한적인 리소스를 가지는 모바일 디바이스에서 동작 가능한 경량 추론 엔진의 개발과 사용자의 이동에 따라 주변 환경에 대한 사전 정보가 없는 상태에서 수집된 주변의 컨텍스트 정보로부터 적절한 서비스를 제공하기 위한 추론의 추상화 및 구체화에 관한 기술의 개발이 요구된다. 즉 발표자에게 발표 자료를 공유할 수 있는 적절한 디스플레이 장치를 추천하는 서비스가 있다고 할 때, 발표자가 새로운 강의실로 이동한 경우 “빔프로젝터” 또는 “프로젝션 TV”에 대한 추론이 아닌 “디스플레이 디바이스”를 검색하는 것이 효율적일 수 있다.

이러한 추상화된 추론을 위해서는 추론 대상에 대한 일반화된 계층적 분류체계(hierarchical taxonomy)가 필수적이다.

본 논문에서는 개인형 모바일 디바이스에서 유비쿼터스 지능형 서비스를 구현하기 위한 핵심 기술인 상황인지 추론 엔진의 구현 방법에 관해 논한다. II장에서는 상황인지 서비스의 구현을 위해 규칙기반추론 엔진을 선택한 이유와 대표적인 규칙기반 추론엔진인 Rete 알고리즘에 대한 관련연구를 기술하고, III장에서는 Rete 알고리즘의 확장상을 통한 향상된 추론 알고리즘을 설명한다.

II. 관련 연구

1. 규칙기반 추론엔진

그림 1은 규칙기반 추론의 기본 사이클을 보여준다. 규칙기반 추론의 기법은 만족해야 하는 조건을 정하고 지정된 조건이 만족(Match)하는 규칙들을 찾아 최선의 규칙을 선택(Select 또는 Conflict resolving)하여 실행(Execute)하며 실행은 새로운 조건의 만족(Match)를 발

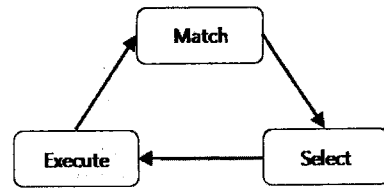


그림 1. 규칙기반 추론의 기본 사이클
Fig. 1. The basic cycle of rule-based reasoning.

생시킬 수 있다.

이러한 방식은 주어진 상황에 대해 적절한 서비스를 선정하는 상황인지기반 서비스의 구현에 적합한 형태를 가지고 있다. 이러한 특성 때문에 최근의 상황인지 연구에 규칙기반 추론 기법이 사용되고 있다.

기존의 규칙기반 추론엔진으로는 CLIPS와 JESS등이 있으며 국내에서는 한국전자통신연구원에서 개발한 Bossam이 있다. CLIPS (C Language Implementation Production System)^[4]는 1985년 NASA의 Johnson Space Center내의 인공지능 연구실에서 개발된 것으로 C언어 기반의 대표적인 규칙기반의 추론엔진이다. 전체적으로 사용되는 명령어나 규칙의 형태는 LISP언어와 비슷하며, Rete 알고리즘을 기반으로 구현되었다. JESS (Java Expert System Shell)^[5]는 CLIPS의 발전된 버전으로 샌디아 국립연구소의 Ernest Fiedman-Hill에 의해 개발되었다. Jess 역시 Rete 알고리즘을 기반 구현되었으며 자바언어가 갖는 이점을 모두 가지면서 표현 스타일은 LISP 언어를 따르는 형태를 취하고 있다. 또한 Jess는 자바의 스크립트적인 요소를 활용하여 자바의 오브젝트의 생성이나 메소드 호출을 컴파일 없이 실행시간에 수행할 수 있다. 국내연구진에 의해 개발된 Bossam^[6] 역시 Rete 알고리즘을 기반으로 하며 Java로 개발되었으며 OWL추론, URI참조, 자바 API를 지원하는 등 기존 상용 규칙 엔진의 주요 기능들을 지원한다.

기존의 규칙기반 추론시스템들은 조건 매칭에 효율적인 Rete 알고리즘을 기반으로 구현되어 효율적인 추론을 수행하며 편리한 사용자 인터페이스를 제공하는 등의 장점을 가지고 있다. 하지만 이러한 시스템들은 전문가 시스템이나 시멘틱웹과 같은 응용을 위해 개발되어 리소스가 충분하지 않은 개인형 모바일 디바이스 환경에서는 사용할 수 없다.

2. Rete 알고리즘

Rete 알고리즘은 규칙기반 추론엔진을 위한 효율적인 패턴매칭 알고리즘이다^[7~8]. 1974년 카네기멜론대학의 Charles L. Forgy 박사에 의해 처음 소개된 이후

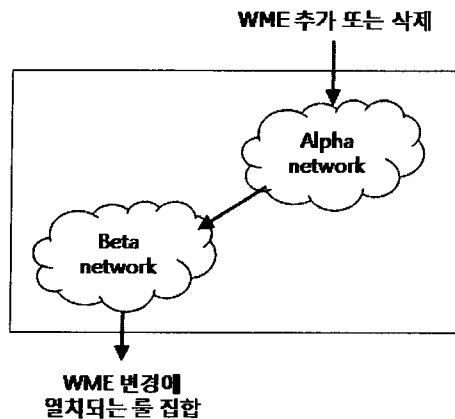


그림 2. Rete 알고리즘의 동작 개요
Fig. 2. The overview of the Rete algorithm.

CLIPS, Jess, JBoss Rules를 비롯한 다수의 널리 알려진 전문가 시스템(expert system)의 기본이 되었다.

Rete 알고리즘은 룰 집합을 메모리상에 네트워크 형태로 구축함으로써 작업 메모리(working memory) 공간의 현재 상태와 룰의 조건(condition 또는 pattern)간의 매칭 검사를 효율적으로 수행한다. 그림 2는 Rete 알고리즘의 동작 개요를 보여준다. 고차원 규칙(rule)을 알파 네트워크와 베타 네트워크로 불리는 연관된 네트워크로 구성되며 구성된 네트워크는 현재 상태의 작업 메모리 공간(current working memory) 상태에서의 WME(Working Memory Element)들에 대한 매칭 상태를 유지한다. 새로운 WME의 추가 또는 기존의 WME의 삭제는 변경된 작업 메모리 공간 상태에 일치되는 룰의 집합을 반환하며 충돌 해결 방법에 따라 선택된 룰이 실행된다.

가. 알파 네트워크(Alpha network)

알파 네트워크는 하나의 WME에 대한 상수 비교 검

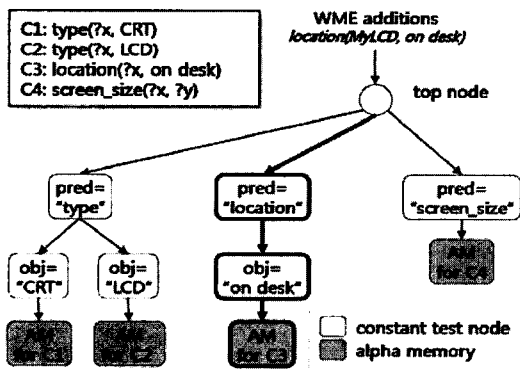


그림 3. Rete 알파 네트워크 구성의 예
Fig. 3. An example of the Rete alpha network.

사를 수행한다. 그림 3은 Rete 알파 네트워크의 구성 예와 새로운 WME "location(MyLCD, on desk)"가 추가된 경우 알파 네트워크에서의 상수 비교 검사를 통해 알파 메모리(alpha memory) "AM for C3"가 활성화되는 과정을 보여준다.

나. 베타 네트워크(Beta network)

베타 네트워크는 하나 이상의 WME 사이의 결합(join)을 수행한다. 그림 4는 Rete 베타 네트워크의 구성 예와 조건 C1, C3에 의해 프로덕션 노드 P1이 활성화 되는 과정을 보여준다.

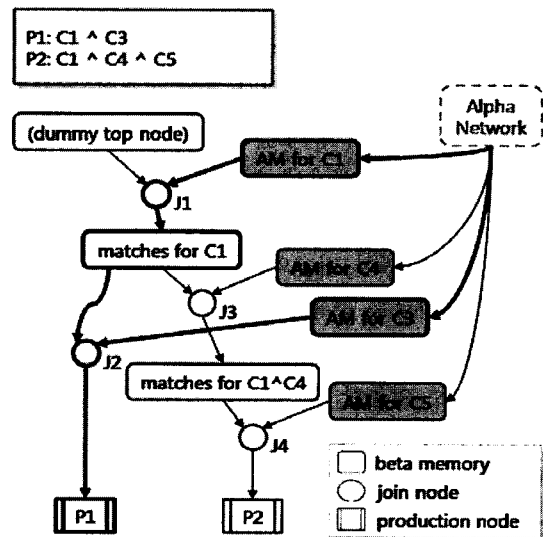


그림 4. Rete 베타 네트워크 구성의 예
Fig. 4. An example of the Rete beta network.

III. Rete 알고리즘 확장

본 논문에서는 개인형 모바일 디바이스에서 지능형 상황인지 서비스를 구현하기 위한 핵심 기술로서 계층

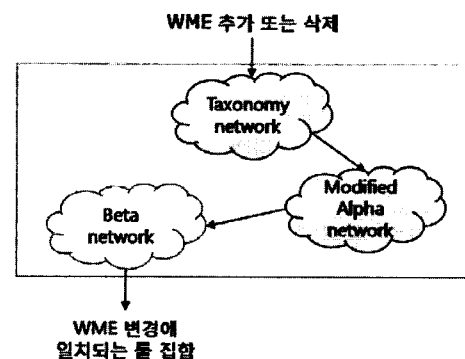


그림 5. Rete 알고리즘 확장의 동작 개요
Fig. 5. The overview of the Rete extension.

적 분류기법을 지원하는 규칙기반 추론엔진을 구현하기 위한 개선된 Rete 알고리즘을 제안한다. 제안된 알고리즘은 서비스 대상 도메인에 대한 컨텍스트정보의 계층적 분류 모델(taxonomy schema)를 표현하는 분류 네트워크(taxonomy network)를 구축하고 Rete 알파 네트워크를 수정하여 기존의 Rete 알고리즘에 계층적 분류에 대한 추론 기능을 추가하였다.

1. 계층적 분류체계 지원

일반적인 룰 기반의 추론 엔진은 현재의 컨텍스트를 표현하는 팩트(fact)와 현재의 컨텍스트가 특정 조건을 만족하는 경우 수행할 작업을 정의하는 규칙(rule)을 정의함으로써 현재의 컨텍스트로부터 새로운 컨텍스트를 추론하는 방식을 취한다.

팩트는 하나의 사실을 표현하기 위한 수단으로 주체(subject), 술어(predicate), 객체(object)로 구성된다. 그림 6은 CLIPS, Jess 등 기존의 룰 기반 추론 엔진들에서 주로 사용되는 표현 방법으로 "seoul"로부터 "youngin"으로의 "path"가 존재함을 나타낸다.

규칙은 실행될 조건을 표현하는 LHS(Left Hand Side)와 조건이 만족되었을 때 수행할 작업의 내용을 표현하는 RHS(Right Hand Side) 부분으로 구성된다. 즉, "LHS의 조건이 만족되면 RHS의 작업이 수행된다(Fired 또는 Triggered)"와 같은 형식을 취한다.

다양한 유비쿼터스 지능 공간으로부터 수집되는 컨텍스트 정보의 모호성을 극복하고 보다 정확한 상황 인지를 수행하거나 모바일 환경에서 추상화된 고차원 추론을 통해 광범위한 추론을 수행하기 위해서는 컨텍스트에 대한 표준 분류 기법(taxonomy) 및 분류된 컨텍

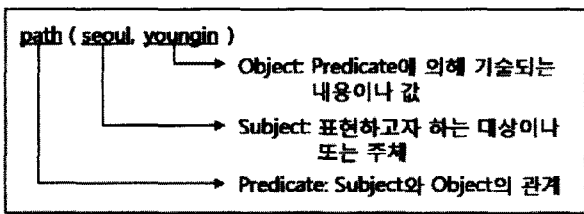


그림 6. 팩트의 표현 예
Fig. 6. An example of the fact expression.

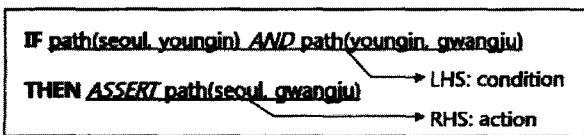


그림 7. 규칙의 표현 예
Fig. 7. An example of the rule expression.

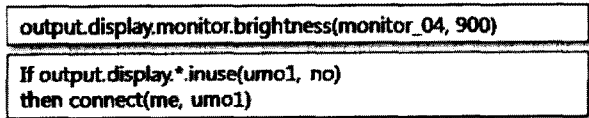


그림 8. 계층적 분류체계에 따른 팩트와 룰의 표현 예
Fig. 8. Representations of a fact and a rule with hierarchical taxonomy expression.

스트 정보를 기반으로 하는 추론 기술이 요구된다. 분류 정보를 활용한 클래스 상관관계 정보의 사용은 일반적인 룰(general rule) 또는 구체적인 룰(specific rule)의 선택적인 구성을 가능하게 함으로써 다양한 수준의 컨텍스트가 실시간으로 수집되는 상황인지 컴퓨팅에서는 중요한 역할을 수행한다. 즉, XPath^[9]와 유사한 wild card ("/", "*")를 이용한 검색 기법을 이용함으로써 기존의 룰 기반 추론엔진들이 가지는 의미론적 추론의 한계를 극복할 수 있다. 그림 8은 계층적 분류체계에 따른 팩트와 룰의 표현 예를 보여준다.

2. Rete 알파 네트워크 확장

Rete 알파 네트워크는 하나의 WME에 대한 필드의 상수 매칭을 검사하는 "constant test node"와 상수 매칭을 통과한 WME들의 저장소인 "alpha memory"로 구성된다. 우리는 기존의 Rete 알파 네트워크에 "constant test node"의 의미를 확장한 "taxonomy test node"를 추가하였다. "taxonomy test node"는 WME와 의 매칭 검사를 위해 단순 상수 비교가 아닌 계층적 분

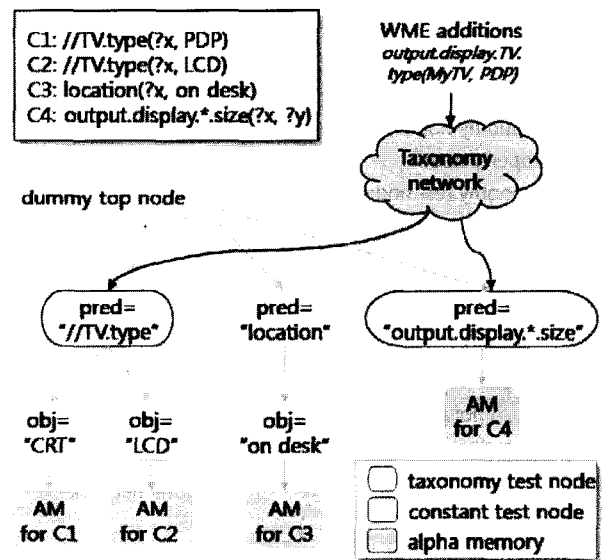


그림 9. Rete 알파 네트워크의 확장 예
Fig. 9. An example of the Rete alpha network extension.

류체계 검색이 수행되어야 함을 의미한다. 계층적 분류체계 검색은 분류 네트워크를 통해 이루어지며, 분류 네트워크의 검색 결과에 따라 "taxonomy test node"가 활성화된다.

그림 9는 C1~C4의 네 가지 조건에 대한 확장된 알파 네트워크의 구성 예를 보여준다. 조건 C1은 "//TV.type"에 대한 분류 검사를 위한 "taxonomy test node"와 객체의 값을 "PDP"와 비교하기 위한 "constant test node"로 구성되며, "output.display.TV.type(MyTV, PDP)"와 같은 새로운 WME에 반응하여 분류 네트워크로부터 활성화된다.

3. Taxonomy 네트워크

계층적 분류체계에 따른 검색을 위해서는 추론 대상 도메인에 대한 분류체계 스키마에 대한 공유가 필요하다. 따라서 추론 엔진은 추론 대상 영역에 대한 스키마를 모델링하고 이를 이용해야 한다. 본 논문에서는 트리형태의 분류체계 스키마를 사용하며 분류 네트워크(taxonomy network)으로 명명한다.

그림 10은 유비쿼터스 스마트 공간에서의 자원 공유를 위한 스키마를 모델링한 분류 네트워크의 일부분으로 분류 체계에 따른 네트워크의 구성과 규칙(Rule)의 조건부에 포함된 분류 검색 표현과의 참조 관계를 보여준다.

위 그림으로부터 알파 네트워크에 추가된 "output.display.*.size" 검색 표현에 대해 "output/display/monitor/size"와 "output/display/TV/size"가 참조관계에 있음을 알 수 있다.

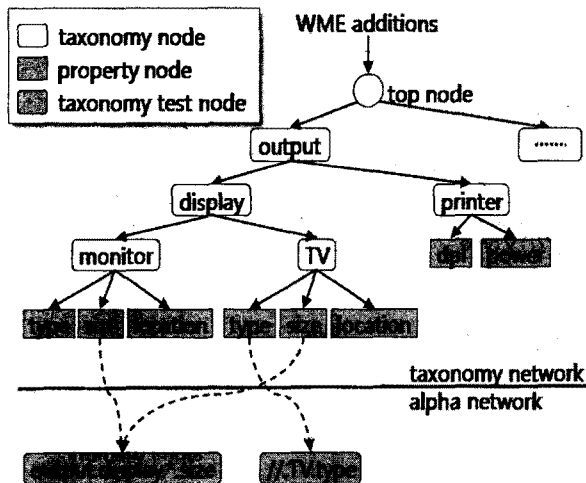


그림 10. Rete 알파 네트워크 구성의 예
Fig. 10. An example of the Rete alpha network.

이 네트워크는 분류 표현을 포함하는 WME로부터 수정된 Rete 알파 네트워크의 "taxonomy test node"를 매핑시키는 기능을 수행한다. 즉, "taxonomy network"에서 WME에 포함된 분류 표현에 일치하는 "property node"를 검색한 후 "property node"에 연결된 모든 "taxonomy test node"를 활성화시킴으로써 추론을 수행한다.

4. 알고리즘 확장에 따른 변경

가. 규칙의 추가 및 삭제

규칙이 추가되면 Rete는 규칙의 해석을 통해 Rete 네트워크의 구조를 변경시킨다. 확장된 Rete 알고리즘 역시 새로운 규칙의 추가에 따라 네트워크의 구조를 변경한다. 이때 규칙의 조건에 포함된 와일드카드 패턴의 분류 표현은 "constant test node"가 아닌 "taxonomy test node"로 생성하고 분류 네트워크에서 검색된 모든 "property node"에 생성된 "taxonomy test node"에 대한 참조를 추가한다.

기존의 규칙이 삭제되는 경우에도 Rete 알고리즘의 고유한 처리 절차의 수행과 분류 네트워크와 Rete 알파 네트워크 간의 참조 수정 작업을 병행해야 한다.

나. WME의 추가 및 삭제

확장된 Rete에서의 WME 추가는 Rete 알파 네트워크가 아닌 분류 네트워크가 된다. 분류 네트워크는 추가된 WME에 일치하는 "property node"를 검색하여 연관된 Rete 알파 네트워크의 "taxonomy test node"를 활성화시키며 이후의 처리 절차는 Rete 고유의 절차와 동일하다.

표 1. Rete네트워크 수정에 따른 추가적인 처리절차
Table 1. Additional procedures for the Rete network extension.

연산	추가 작업
초기화	분류 구조(taxonomy schema)정보로부터 분류 네트워크를 구성
규칙의 추가	Rete 알고리즘 고유 작업
규칙의 삭제	+ 분류 네트워크와 알파 네트워크간의 참조 갱신
WME의 추가	새로운 WME의 진입점의 변경(알파 네트워크로부터 분류 네트워크로)
WME의 삭제	변경 없음

IV. 성능평가

본 논문에서는 모바일 환경에 최적화된 상황인지 엔진의 개발을 위한 확장된 Rete 알고리즘을 제안하고 ARM 11 프로세서, 64MB RAM, 128MB 플래쉬 메모리로 구성된 모바일 플랫폼에서 구현하였다. 전방향(forward chaining) 추론 엔진은 메모리 사용량이 많기 때문에 자원이 부족한 모바일 환경에서 사용이 어렵거나 불가능하다.

본 구현의 메모리 사용량의 비교 평가를 위해서는 Rete 알고리즘을 기반으로 구현된 CLIPS와 Jess를 사용하였으며 동일한 서비스 시나리오 구현에 따른 동작 메모리 사용량을 단계별로 비교하였으며, 비교를 위해

```

Rete 확장
IF OutputDevice.Display.*.size(?entity_id,?size),
  comparison:>(?size,1)
THEN
  print(?entity_id)

CLIPS/Jess
(defrule MyRecommendationMonitor1
  (IsA (Parent OutputDevice)(Child ?display))
  (IsA (Parent ?display)(Child ?device))
  (test (str-compare ?device Monitor))
  (Monitor (ID ?id)(size ?size))
  (test (> ?size 1))
=>
  (printout t ?id crlf))
(defrule MyRecommendationMonitor2
  (IsA (Parent OutputDevice)(Child ?display))
  (IsA (Parent ?display)(Child ?device))
  (test (str-compare ?device PDP))
  (PDP (ID ?id)(size ?size))
  (test (> ?size 1))
=>
  (printout t ?id crlf))
  
```

그림 11. 시나리오 구현을 위한 규칙 정의
 Fig. 11. A rule definition for implementing the given scenario.

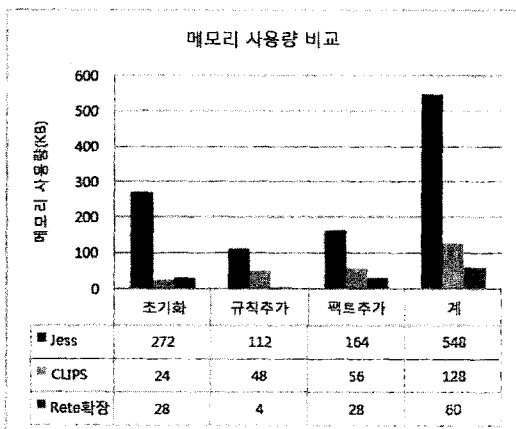


그림 12. 메모리 사용량 비교
 Fig. 12. A comparison result about memory usage.

“주변의 디스플레이 장치 중 화면 크기가 30cm 이상인 것을 추천”하는 서비스에 대한 규칙을 각 시스템에 맞게 작성하였고, 55개의 팩트(Fact)를 사용하였다. 메모리 사용량의 측정은 초기화 단계, 규칙의 추가 단계, 팩트의 추가 단계에서 수행하였다.

위의 서비스 구현을 위해 각 시스템에서 사용된 규칙은 그림 11과 같다. 제안된 알고리즘은 계층구조 검색을 지원하기 때문에 하나의 규칙만으로 정의가 가능한 반면 Jess와 CLIPS의 경우 계층 구조 검색을 복잡한 형태의 조건 정의가 필요하며 추론의 일반화(와일드 카드 검색)를 위해 하나 이상의 규칙 정의가 필요하다.

그림 12는 측정 결과를 보여준다. Jess는 자바로 구현되어 자바 버추얼 머신이 기본적으로 유지하는 메모리 및 규칙과 팩트의 추가에 따른 메모리 증가가 큰 것으로 나타났으며, CLIPS는 비교적 작은 메모리를 사용하지만 분류 체계 검색 지원을 위해 추가적인 규칙을 정의함에 따라 많은 메모리를 사용하고 있음을 알 수 있다. 결과를 통해 본 논문에서 제안하고 있는 확장 Rete 알고리즘이 주어진 시나리오에 효과적인 규칙의 생성에 따른 제한된 메모리 환경에서의 효율적인 운용이 가능함을 알 수 있다.

V. 결론

본 논문에서는 모바일 환경에서의 지능형 서비스를 구현하기 위한 상황인지 엔진의 구현에 관해 기술하였다. 기존의 추론 기법 및 추론 엔진들에 대한 조사를 기반으로 규칙기반 추론 기법인 Rete 알고리즘을 확장함으로써 유비쿼터스 모바일 디바이스 환경에 특화된 상황인지 추론엔진을 설계 및 구현하였다.

특히 모바일 환경에서의 효율적인 추론을 위해 제안된 계층적 분류체계 지원을 통한 추론의 추상화 및 일반화 기법은 기존의 규칙기반 추론 엔진의 기능 확장을 통해 고차원 추론 엔진을 구현함으로써 모바일 커뮤니티 컴퓨팅 환경에서의 다양한 신규 서비스를 창출하는데 기여할 수 있을 것으로 기대된다.

참고 문헌

[1] H. Chen et al., Intelligent Agents Meet Semantic Web in a Smart Meeting Room. 3rd International Joint Conference on Autonomous Agents and Multiagent Systems, pp. 854-861,

July, 2004.

[2] H. Chen, T. Finin, and A. Joshi. A context broker for building smart meeting rooms. In Proceedings of the Knowledge Representation and Ontology for Autonomous Systems Symposium, 2004 AAAI Spring Symposium. AAAI, March 2004.

[3] Jih, W. R.; Cheng, S. Y.; and Hsu, J. Y. J. Context-aware access control on pervasive healthcare. In *EEE'05 Workshop: Mobility, Agents, and Mobile Services (MAM)*, 2005.

[4] G. Riley. CLIPS-A Tool for Building Expert Systems. <http://www.ghg.net/clips/CLIPS.html>, 2003.

[5] Friedman-Hill, Ernest J., JESS, Java Expert System Shell, Sandia Nation Laboratories, Livermore, CA, <http://herzberg.ca.sandia.gov/jess>.

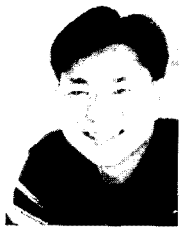
[6] <http://projects.semwebcentral.org/projects/bossam/>

[7] Forgy, C. Rete, A Fast Algorithm for the Many Patterns/Many Objects Match Problem. *Artificial Intelligence*, 19(1), 17-37, 1982.

[8] Doorenbos, R. B. Production Matching for Large Learning Systems, Doctoral dissertation, Computer Science Department, Carnegie Mellon Univ., 1995.

[9] Berglund, A., Boag, S., Chamberlin, D., Fernandez, M. F., Kay, M., Robie, J., and Siméon, J. 2002. XML Path Language (XPath) 2.0. Tech. Rep. W3C Working Draft, Version 2.0, World Wide Web Consortium. Aug. <http://www.w3.org/TR/xpath20/>.

— 저 자 소 개 —



김 태 현(정회원)
 1999년 한국외국어대학교 컴퓨터 공학과 학사 졸업.
 2001년 충남대학교 컴퓨터공학과 석사 졸업.
 2008년 현재 전자부품연구원 유비쿼터스연구센터 연구원.

2008년 현재 한국외국어대학교 컴퓨터공학과 박사과정
 <주관심분야 : WSN, 상황인지>



김 재 호(정회원)
 1996년 한국외국어대학교 컴퓨터 공학과 학사 졸업.
 2000년 한국외국어대학교 컴퓨터 공학과 석사 졸업.
 2008년 현재 전자부품연구원 유비쿼터스연구센터 선임연구원.

<주관심분야 : 유비쿼터스 센서 네트워크, 센서네트워크 기반 상황인지>



원 광 호(정회원)
 1981년 단국대학교 전자공학과 학사 졸업.
 2004년 중앙대학교 정보통신 공학과 석사 졸업.
 2007년 광운대학교 전자통신 공학과 박사 수료.

2008년 현재 전자부품연구원 유비쿼터스연구센터 책임연구원.
 <주관심분야 : WPAN(ZigBee, UWB) 기술, 지능형 홈 네트워크>



이 기 혁(학생회원)
 2007년 한국외국어대학교 컴퓨터 공학과 학사.
 2008년 현재 한국외국어대학교 컴퓨터공학과 석사 재학.
 <주관심분야 : 데이터베이스, 유비쿼터스>



손 기 락(정회원)
 1984년 서울대학교 계산통계학과 학사 졸업.
 1986년 서울대학교 계산통계학과 석사 졸업.
 1993년 미국, Univ. of California, Santa Cruz, 전산학 박사.
 1994년~1996년 전자통신연구원 선임연구원.
 1996년~현재 한국외국어대학교 컴퓨터및정보통신공학부 교수.
 <주관심분야 : 데이터베이스, 데이터마이닝>