

다중추론지원 분산형 상황인식 시스템을 위한 통합 상황모델

정장섭*, 홍승택*, 장대진*, 방대욱*

*계명대학교 컴퓨터공학과

e-mail:hdca64@hanafos.com*, nawoori7829@gmail.com*, djjang@kmu.ac.kr*,
dubang@kmu.ac.kr*

An Unified Context Model for A Context-Aware System Supporting Distributed Processing and Multi-Reasoning

Jang-Seop Jeong*, Seung-Taek Hong*, Dae-Jun Jang*, Dae-Wook Bang*

*Dept of Computer Engineering, Keimyung University

요 약

본 논문에서는 모바일 컴퓨팅 환경과 불확실성을 지원하는 다중추론지원 분산형 상황인식 시스템의 지식 베이스(KB: Knowledge Base)를 위한 모델로써 상황정보(OWL), 온톨로지 추론정보(OWL DL), 규칙 추론정보(SWRL), 베이지안 추론정보(PR-OWL)를 통합적으로 표현하는 UniOWL 통합상황모델을 제안한다. 제안한 통합상황모델은 상황정보와 다중 추론정보를 단일 구문, 즉 OWL 구문으로 표현하여 지식베이스 설계를 수월하게 하고 표현을 단순화하는 장점이 있다.

1. 서 론

최근 모바일 컴퓨팅 환경 지원을 실현하기 위한 연구가 전 세계적으로 급증함에 따라, 그 중요성은 점점 더 증가하고 있다[1]. 모바일 컴퓨팅 환경에서 지능형 서비스를 제공하기 위해서는 상황 데이터를 수집하고 적절한 가공을 통해 상황정보로 변환시켜 해석, 추론 및 학습 과정을 거쳐 사용자의 상황에 맞는 적절한 서비스를 제공하는 모바일 상황인식시스템이 필요하다[2].

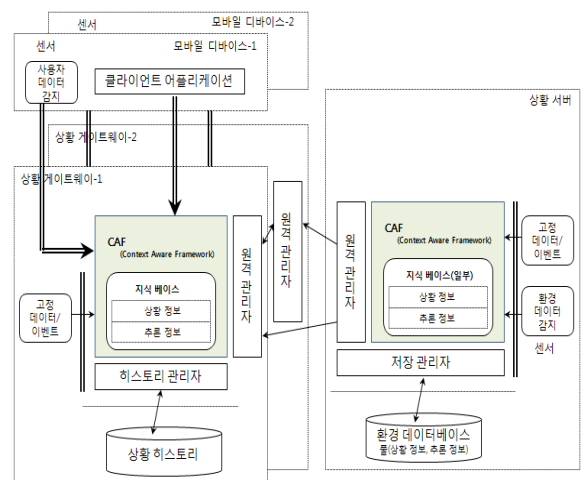
상황인식 시스템은 상황을 시스템 내에 적절하게 표현할 상황모델을 필요로 한다. 현재까지 상황을 모델링하는 방법으로 키-값 모델(Key-Models), 마크업 스키마 모델(Markup Scheme Models), 그래픽 모델(Graphical Models), 객체 지향모델(Object Oriented Models), 논리 기반 모델(Logic Based Models), 온톨로지 기반 모델(Ontology Based Models) 등이 사용 되었다[3].

본 논문에서는 모바일 컴퓨팅 환경과 불확실성을 지원하는 다중추론지원 분산형 상황인식 시스템의 지식 베이스(KB-Knowledge Base)를 위한 모델로써 상황정보(OWL), 온톨로지 추론정보(OWL DL), 규칙 추론정보(SWRL), 베이지안 추론정보(PR-OWL)를 통합적으로 표현하는 통합 상황모델(UniOWL)을 제안한다.

2. 다중추론지원 분산형 상황인식 시스템

다중추론지원 분산형 상황인식 시스템은 모바일 상황인식 환경에서 사용자의 상황을 고려하여 지능적인 서비스를 제공하는 시스템이다[4]. 이 시스템은 (그림 1)과 같이 3계층 구조로 구성되는데, 3계층 구조는 2계층 구조보다 모바일 디바이스의 처리 부담을 경감시켜 모바일 디바이스가 고성능 처리능력을 가지지 않아도 상황인식을 기반

으로 한 지능형 서비스가 가능한 구조이다.



(그림 1) 다중추론지원 분산형 상황인식 시스템 구조

3계층 구조의 상황인식시스템은 상황 게이트웨이와 상황 서버에 상황인식 프레임워크에 해당하는 CAF(Context Aware Framework)를 두는 분산형 상황인식 프레임워크를 사용한다. 그리고 시스템의 모든 구성요소들은 유무선 네트워크로 연결되어 센서 정보, 환경 정보, 추론 정보 등을 교환하고 공유한다.

모바일 디바이스(Mobile Device)는 CAF(Context Aware Framework)에 의해 관리되고 추론된 상황정보를 사용하는 클라이언트 어플리케이션을 수행한다. 클라이언트 어플리케이션은 CAF API(Application Program Interface)로 CAF에 접근하여 상황정보를 활용하고, 모바일 디바이스의 UI(User Interface)를 통해 사용자와 상호

작용한다. 그리고 모바일 디바이스(Mobile Device)에 장착된 센서로 감지되는 사용자 개인정보를 실시간으로 게이트웨이에 있는 CAF로 보내 CAF가 항상 현재 시점의 상황정보를 관리할 수 있도록 한다.

상황 게이트웨이(Context Gateway)는 모바일 단말기별로 존재하면서 사적 상황정보를 관리한다. 상황 게이트웨이의 CAF는 지식베이스로 사적 상황정보를 저장 및 관리하고, 다중 상황추론기로 논리적인 상황정보도 생성 및 저장하며, 지식 베이스에 있는 물리적 상황정보(Fact)와 논리적인 상황정보를 클라이언트 어플리케이션에게 제공한다. 원격 관리자(Remote Manager)는 상황정보 공유를 필요로 하는 상황 게이트웨이들과 상황서버를 연결하는 상황 네트워크를 구성하고, 구성요소들이 실시간으로 분산된 상황정보를 공유할 수 있게 한다. 히스토리 관리자(History Manager)는 시점 변화에 따라 발생하는 과거 상황정보를 기록하여 현재 뿐만 아니라 과거 상황정보도 클라이언트 어플리케이션에 제공할 수 있게 한다.

상황 서버(Context Server)는 모든 사용자들이 공용으로 사용하는 환경 상황정보를 관리한다. 상황 서버의 CAF는 외부 환경에서 감지되는 환경 상황정보를 수집하여 지식베이스(KB)에 저장하고, 지식 베이스에 저장된 상황정보를 사용한 다중추론기로 추론하여 새로운 상황정보도 생성 및 저장한다. 저장 관리자(Store Manager)는 지식 베이스와 환경 데이터베이스에 대하여 동기화를 수행하여 대용량의 환경 상황정보 중에서 현재 사용 중인 정보만 지식 베이스로 운용되도록 자동 관리를 한다.

CAF는 상황 수집자, 쿼리 프로세서, CAF 코어모듈, 그리고 지식베이스로 구성된다. 상황 수집자는 외부로부터 수집된 고정 데이터와 사용자 데이터를 CAF 코어모듈을 사용하여 지식베이스에 저장하며, 필요시 추론 엔진을 가동하여 상황정보를 추론한다. 쿼리 프로세서는 클라이언트 어플리케이션이 CAF API를 통해 전달하는 SPARQL 또는 SQWRL 구문으로 작성된 쿼리를 파싱하고, CAF 코어모듈을 사용하여 지식베이스에서 요구된 상황정보를 검색하여 제공한다. CAF 코어모듈은 내부에 추론엔진을 두어 온톨로지 추론, 규칙 추론, 베이지안 추론을 순차적 또는 선택적으로 수행하는 다중 상황추론을 담당하고, 지식베이스에서 지식을 추출, 갱신, 무효화를 담당한다.

CAF는 ready, run, block 상태를 가지며, CAF에 발생되는 이벤트에 의해 상태가 전이된다. 최초 구동시 초기화에 해당되는 initialize 이벤트로 ready 상태가 시작되며, ready 상태는 외부 요구로 발생하는 start 이벤트로 run 상태가 된다. run 상태는 외부 발생 stop 이벤트로 block 상태로 전환되어 일시 중지되어 있다가 resume 이벤트로 다시 run 상태가 된다.

지식베이스는 상황모델, 상황정보, 그리고 추론메타정보로 구성되어 있다. 상황모델은 OWL 온톨로지로 상황 실체(entity)를 정의하고 실체 관계도(relationship)를 표현한다. 상황정보는 사실(물리적 정적 및 동적 데이터)과 논리 정보(추론에 의해 생성된 정보)를 포함하고, OWL 인스턴스로 표현된다. 추론메타정보는 DL 연산자(operator), 규칙정보, 베이지안 네트워크를 포함한다. 규칙정보는 OWL로 표현된 상황정보와 함께 표현할 수 있게 하는 SWRL

구문으로 작성된다. 또한 베이지안 네트워크는 상황정보 및 규칙정보와 함께 표현할 수 있도록 하는 OntoBayes, BayesOWL, 또는 PR-OWL로 표현된다.

3. 통합 상황모델

3.1 통합 상황모델 구성과 표현

통합상황모델 UniOWL은 상황정보와 추론정보를 함께 표현하며, 다양한 추론 정보들을 통일된 표현, 즉 OWL로 표현한다. 본 논문에서 제안하는 통합 상황모델, 즉 UniOWL은 (그림 2)의 구문으로 정의된다.

```
<owl:Class rdf:ID="상황정보">
<owl:DatatypeProperty rdf:ID="소스경로">
  <rdfs:domain rdf:resource="#상황정보"/>
  <rdfs:range rdf:resource="xsd:string"/> /* 소스 URI */
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="소스유형">
  <rdfs:domain rdf:resource="#상황정보"/>
  <rdfs:range rdf:resource="xsd:string"/> /*all,model,data*/
</owl:DatatypeProperty>

<owl:ObjectProperty rdf:ID="상황추론관계">
  <rdfs:domain rdf:resource="#상황정보"/>
  <rdfs:range rdf:resource="#추론정보"/>
</owl:ObjectProperty>

<owl:Class rdf:ID="추론정보">
<owl:Class rdf:ID="온톨로지추론정보">
  <rdf:subClassOf rdf:resource="#추론정보"/>
</owl:Class>
<owl:Class rdf:ID="규칙추론정보">
  <rdf:subClassOf rdf:resource="#추론정보"/>
</owl:Class>
<owl:Class rdf:ID="베이지안추론정보">
  <rdf:subClassOf rdf:resource="#추론정보"/>
</owl:Class>
<owl:DatatypeProperty rdf:ID="추론시점">
  <rdfs:domain rdf:resource="#추론정보"/>
  <rdfs:range rdf:resource="xsd:string"/> /* ready ,run, block */
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="추론소스표현">
  <rdfs:domain rdf:resource="#추론정보"/>
  <rdfs:range rdf:resource="xsd:string"/> /* OWL, SWRL,
  PRO-OWL */
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="추론소스경로">
  <rdfs:domain rdf:resource="#추론정보"/>
  <rdfs:range rdf:resource="xsd:string"/> /* 소스 URI */
</owl:DatatypeProperty>
```

(그림 2) UniOWL 표현 통합상황모델

UniOWL에서 상황정보는 소스경로에 표기된 위치에 있는 OWL 문서이다. 그리고 추론정보는 온톨로지 추론정보, 규칙 추론정보, 베이지안 추론정보로 세분화 되고, 각 추론정보는 추론시점별로 적용될 소스경로로 표현된 위치의 OWL 문서이다.

3.2 상황정보 표현

UniOWL의 상황정보 소스경로에 표기된 위치의 OWL 문서는 상황모델의 개체(entity)를 Class로 표현하고, 개체 관계 및 속성을 ObjectProperty와 DataTypeProperty로 표현한다. (그림 3)은 상황정보를 OWL 구문으로 표현한 사례이다.

```

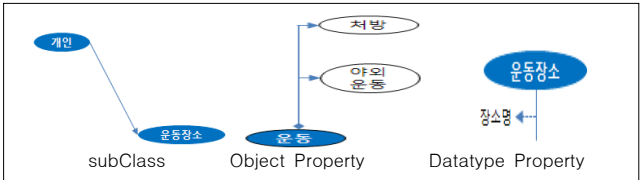
<owl:Class rdf:ID="개인">
</owl:Class>
<owl:Class rdf:ID="운동">
</owl:Class>
<owl:Class rdf:ID="운동장소">
</owl:Class>
<owl:Class rdf:ID="운동기록">
</owl:Class>

<owl:Class rdf:ID="처방">
  <rdf:subClassOf rdf:resource="#운동" />
</owl:Class>
<owl:Class rdf:ID="야외운동">
  <rdf:subClassOf rdf:resource="#운동" />
</owl:Class>

<owl:ObjectProperty rdf:ID="has_운동장소">
  <rdfs:domain rdf:resource="#개인" />
  <rdfs:range rdf:resource="#운동장소" />
<owl:inverseOf rdf:resource="#운동장소_in" />
</owl:ObjectProperty>

<owl:DatatypeProperty rdf:ID="장소명">
  <rdfs:domain rdf:resource="#운동장소" />
  <rdfs:range rdf:resource="xsd:string" />
</owl:DatatypeProperty>
    
```

(a) 상황정보 OWL 표현



(b) 상황정보 그래픽 표현

(그림 3) 상황정보의 표현 사례

3.3 추론정보 표현

UniOWL의 추론정보는 온톨로지 추론정보, 규칙 추론정보, 페이지안 추론정보로 세분화된다.

(1) 온톨로지 추론정보

온톨로지 추론정보는 OWL-DL의 연산자 구문이다. 대표적인 연산자로는 합집합(owl:unionOf), 교집합(owl:intersectionOf), 여집합(owl:complementOf), 열거(owl:oneOf) 등이 있다. (그림 4)는 온톨로지 추론정보를 표현한 사례이다.

```

(owl:Class rdf:ID="야외운동"
  <owl:unionOf rdf:parseType="Collection">
    <owl:Class rdf:about="#워킹" />
    <owl:Class rdf:about="#하이킹" />
    <owl:Class rdf:about="#종합" />
  </owl:unionOf>
</owl:Class>

<owl:Class rdf:ID="처방">
  <owl:intersectionOf rdf:parseType="Collection">
    <owl:Class rdf:about="#워킹처방" />
    <owl:Class rdf:about="#하이킹처방" />
  </owl:intersectionOf>
</owl:Class>

<owl:Class rdf:ID="Non_휴대장치">
  <owl:complementOf rdf:resource="#휴대장치">
</owl:Class>
    
```

```

<owl:Class rdf:ID="야외운동">
  <owl:oneOf rdf:parseType="Collection">
    <owl:Thing rdf:about="#워킹" />
    <owl:Thing rdf:about="#하이킹" />
    <owl:Thing rdf:about="#종합" />
    
```

(a) OWL 표현

(워킹 U 하이킹 U 종합) -> 야외운동
 (워킹처방 U 하이킹처방) -> 처방
 "휴대장치 -> Non_휴대장치"
 {워킹, 하이킹, 종합} -> 야외운동

(b) 연산자 표현

(그림 4) 온톨로지 추론정보 표현 사례

(2) 규칙 추론정보

규칙 추론정보는 SWRL 구문으로 정의된다. SWRL[5]은 OWL에 규칙 기술 언어를 추가한 구조를 가지며, OWL에서 표현하기 힘든 복잡한 규칙을 정의할 수 있는 언어이다. (그림 5)의 (b)와 같은 서술논리에 해당하는 SWRL 구문은 (a)와 같다.

```

<ruleml:imp>
<ruleml:_rlab ruleml:href="#rule1"/>
<ruleml:_body>
  <swrlx:individualPropertyAtom swrlx:property="고령도">
    <ruleml:var>x1</ruleml:var>
    <ruleml:var>"high"</ruleml:var>
  </swrlx:individualPropertyAtom>
  <swrlx:individualPropertyAtom swrlx:property="질병보유도">
    <ruleml:var>x1</ruleml:var>
    <ruleml:var>"high"</ruleml:var>
  </swrlx:individualPropertyAtom>
<ruleml:_body>
<ruleml:_head>
  <swrlx:individualPropertyAtom swrlx:property="건강도">
    <ruleml:var>x1</ruleml:var>
    <ruleml:var>"low"</ruleml:var>
  </swrlx:individualPropertyAtom>
</ruleml:_head>
</ruleml:imp>
    
```

(a) OWL 표현

고령도(x1, "high"), 질병보유도(x1, "high") -> 건강도(x1, "low");

(b) 서술논리 표현

(그림 5) 규칙 추론의 표현 사례

(3) 페이지안 추론정보

페이지안 추론은 확률기반 추론으로 추론정보의 OWL 표현으로 OntoBayes와 BayesOWL이 있다[6]. OntoBayes는 온톨로지 웹 언어인 OWL을 확장하여 FullProbDist, HasPrior, PriorProb, hasCond, CondProb, dependOn의 태그를 정의하고, 이를 통해 객체 속성(Object Property)이나 데이터 속성(Data Property)간의 의존관계와 조건부 확률 테이블 값을 정의한다. OntoBayes를 이용하면, 에이전트는 자신의 행동이 환경에 미치는 영향을 확률적으로 추론할 수 있다는 장점이 있다. 그러나 OntoBayes는 OWL을 통해서 Bayesian Network의 성질을 표현하였을 뿐, 온톨로지 추론과정에 Bayesian Network가 적용되는 것이 아니기 때문에, Ontology와 Bayesian Network 모델

을 각각 가지고 있는 것과 큰 차이가 없다.

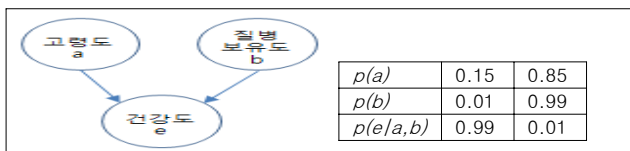
BayesOWL은 OntoBayes와 달리 온톨로지 전체를 베이 지안 네트워크로 변화시켜 온톨로지서 확률 추론이 가 능하다. BayesOWL은 객체나 속성간의 의존관계를 온톨 로지의 구조를 그대로 따라서 정의하며, OntoBayes와 같 이 조건부 확률 값을 위한 몇 가지 태그를 정의하고 있다. 그러나 BayesOWL은 온톨로지 전체를 베이 지안 네트워크 로 구성하므로, 거대하고 복잡한 도메인에 대해서 각 객체 간의 관계를 조건부 확률로 나타내기 위해서는 많은 양의 조건부 확률 값을 설정해 주어야하므로, 초기 설계에 많은 시간이 필요하다. 그리고 학습을 위해서는 온톨로지 전체 에 대한 데이터 수집이 필요하므로 많은 시간과 노력이 필요하며, 따라서 온톨로지 데이터의 확장에 있어서 많은 비용이 필요하다[7].

UniOWL에서 베이 지안 추론정보는 기존 OntoBayes, BayesOWL, PRO-OWL 중에서 선택하여 표현한다. (그림 6은) 고령도와 질병보유도의 evidence가 주어졌을 때 건강 도가 True인 확률이 얼마인지를 조건부 확률을 구하는 베 이 지안 추론정보의 표현 사례를 BayesOWL 표현과 베 이 지안 네트워크 및 조건부 이벤트에 대한 확률을 각각 보 여주고 있다.

```

<Variable rdf:ID="c">
  <hasClass>건강도</hasClass>
  <hasState>True</hasState>
</Variable>
<Variable rdf:ID="p1">
  <hasClass>고령도</hasClass>
  <hasProbValue>True</hasState>
</Variable>
<Variable rdf:ID="p2">
  <hasClass>질병보유도</hasClass>
  <hasState>True</hasState>
</Variable>
<PriorProb rdf:ID="P(c)">
  <hasVariable>p1</hasVariable>
  <hasProbValue>0.15</hasProbValue>
</PriorProb>
<PriorProb rdf:ID="P(c)">
  <hasVariable>p2</hasVariable>
  <hasProbValue>0.01</hasProbValue>
</PriorProb>
<CondProb rdf:ID="P(c|p1, p2)">
  <hasCondition>p1</hasCondition>
  <hasCondition>p2</hasCondition>
  <hasVariable>c</hasVariable>
  <hasProbValue>0.99</hasProbValue>
</CondProb>
    
```

(a) BayesOWL 표현



(b) 베이 지안 그래픽 표현

(그림 6) 베이 지안 추론정보의 표현 사례

4. 결론

본 논문에서는 모바일 컴퓨팅 환경과 불확실성을 지원 하는 다중추론 및 분산처리 상황인식 시스템의 지식 베 이 스를 위한 모델로써 상황정보와 추론정보를 통합적으로 표현한 OWL 기반 통합 상황모델, 즉 UniOWL를 제안하 였다. UniOWL은 모델 설계에 드는 시간 비용을 줄일 수 있을 뿐만 아니라, 손쉽게 네트워크를 설계할 수 있으며, 온톨로지의 확장성을 높여준다.

다중추론지원 분산형 상황인식시스템을 구현하려면, 적 합한 상황모델 연구 외에도 효율적인 추론을 수행하는 추 론기의 구현 연구가 필요하다. 즉 추론시 불필요한 추론규 칩은 처리되지 않도록 하는 추론선택 스케줄링 알고리즘 적용한 추론기 연구가 계속 진행되어야 할 것이다.

참 고 문 헌

[1] G. Specht and T. Weithoner. "Context-Aware Processing of Ontologies in Mobile Environments," 7th International Conference on Mobile Data Management MDM06 (2006) Volume: 2006, Publisher: Ieee, Pages: 86-86.

[2] M. Duckham, et al., Location privacy and location-aware computing, Dynamic & Mobile GIS: Investigating Change in Space and Time, 2006.

[3] H. Kim, Y.J. Cho, and S.R Oh, "CAMUS-A Middleware Supporting Context-aware Services for Network-based Robots," *IEEE Workshop on Advanced Robotics and Its Social Impacts, Nagoya, japan, 2005*.

[4] 정장섭, 방대욱, 다중추론지원 분산형 상황인식 시스템, 한국정보과학회, 2012 한국컴퓨터종합학술대회 논문집 Vol.39, No.1(D) pp. 91-93, 2012.

[5] SWRL (A Semantic Web Rule Language Combining OWL and RuleML), W3C Member Submission 21 May 2004. <http://www.w3.org/Submission/SWRL/>, 2004.

[6] Y. Yang and J. Calmet, "OntoBayes: An ontology-driven uncertainty model," Proc. of the Int. conf. on Intelligent Agents, vol 1, pp. 457-464, 2005.

[7] Kleiter, G. D. (1996) "Propagating imprecise probabilities in Bayesian networks," Artificial Intelligence, Vol. 88, No. 1-2, pp. 143-161.