

A Formal Model and a Design of Inference Engine for Context-Aware Mobile Computing

Moon Kwon Kim[†] · Soo Dong Kim^{††}

ABSTRACT

Context-aware mobile computing has become the primary approach to realize automatic, autonomous, and user-centric computing in the context of largely increasing the amount of mobile devices used that embed available sensors. However, designing an inference engine nonetheless requires the tasks of analyzing contexts, situations that can be inferred, etc. Moreover, a mobile device has limited resources and limited computation capability, which results in recognizing the common sense of its unsuitable environment for processing inference. Hence, we propose context-situation reasoning elements and their formal models in this paper, and we verify the formal models' applicability by applying them to an example. Finally, we design and implement an inference engine that realize the context-situation inference elements in computing environment, and we experiment an example by using the proposed inference engine to verify applicability and reusability of the inference engine.

Keywords : Mobile Computing, Context-Aware Computing, Pervasive Computing, Inference Engine

컨텍스트 인지 모바일 컴퓨팅을 위한 정형모델 및 추론 시스템 설계

김 문 권[†] · 김 수 동^{††}

요 약

가용 센서를 내장하고 있는 모바일 디바이스의 사용이 많아지고 자동화, 자율화, 사용자 맞춤형 서비스의 요구가 커짐에 따라 컨텍스트 인지 모바일 컴퓨팅 (Context-Aware Mobile Computing)의 필요성이 증대하고 있다. 하지만 추론 시스템 설계는 컨텍스트 분석, 인지하고자 하는 상황분석 등의 복잡한 과정을 요구한다. 또한 모바일 디바이스의 제한된 자원 때문에 컴퓨팅 파워가 높은 서버에 탑재된 추론 엔진을 통해 추론을 수행하는 것이 적합하다. 본 논문에서는 이러한 과정을 간결하고 정확하게 표현하기 위한 컨텍스트-상황 추론 요소의 범용적 정형 모델을 제안하고 추론 요소들의 정형 모델을 실사례에 적용하여 본 논문에서 제안하고 있는 추론 요소들의 정형 모델이 실효성을 가지고 있으며 범용적임을 보여준다. 또한 제한된 추론 요소들을 컴퓨팅 환경에서 실현화하기 위해 제안한 정형 모델들을 기반으로 추론 엔진을 설계 및 구현하고 추론 실험을 통해 추론 엔진의 실효성과 재사용성을 검증한다.

키워드 : 모바일 컴퓨팅, 컨텍스트 인지 컴퓨팅, 퍼베이시브 컴퓨팅, 추론 엔진

1. 서 론

컨텍스트 인지 컴퓨팅 (Context-Aware Computing)은 실세계의 사물, 사건 등의 상태를 특징화 및 정보화하여 컴퓨팅 환경에서 실세계의 상황을 인지하기 위한 기술을 의미한다[1]. 최근 자동화, 자율화, 사용자 맞춤형 서비스의 선호가

커짐에 따라 컨텍스트 (Context)를 분석하여 실세계의 상황을 추론 및 활용하는 컨텍스트 인지 컴퓨팅의 필요성이 커지고 있으며 이에 대한 연구가 활발히 진행되고 있다.

최근 가용 센서를 내장하고 있는 모바일 디바이스의 사용이 증가함에 따라 센서로부터 수집되는 컨텍스트의 종류와 양 또한 많아지고 있어, 센서 기반 컨텍스트를 이용한 상황 추론이 가능해지고 있다. 하지만 컨텍스트 인지 모바일 컴퓨팅에서의 컨텍스트-상황 추론은 다양하고 방대한 컨텍스트 분석, 추론하고자 하는 상황 및 도메인 분석, 컨텍스트-상황 추론 과정에 활용될 알고리즘 분석 등의 복잡한 모델링 과정을 요구한다. 이러한 복잡한 컨텍스트-상황 추론 과

※ 본 연구는 방위사업청과 국방과학연구소의 지원으로 수행되었음(UD060048AD).

† 준 회 원: 숭실대학교 컴퓨터학과 석사과정

†† 종 신 화 원: 숭실대학교 컴퓨터학부 교수

논문접수: 2012년 12월 24일

수정일: 1차 2013년 1월 4일

심사완료: 2013년 1월 4일

* Corresponding Author: Soo Dong Kim(sdkim777@gmail.com)

정을 컨텍스트-상황 추론 요소들의 정형 모델을 이용하여 표현하면 추론에 이용될 컨텍스트, 추론하고자 하는 상황 및 도메인, 각 상황을 표현하는 컨텍스트, 추론을 위한 알고리즘 등을 간결하고 명료하게 표현 할 수 있어, 컨텍스트-상황 추론 과정을 보다 효과적으로 명세할 수 있다.

컨텍스트-상황 추론은 높은 CPU, 메모리, 저장소 등의 자원 활용이 필요하다. 하지만 모바일 디바이스는 제한된 자원, 즉, 낮은 컴퓨팅 파워, 적은 메모리와 저장소를 가진다. 또한 모바일 디바이스는 배터리를 통해 전력을 공급 받기 때문에 컨텍스트-상황 추론과 같은 자원 활용이 많은 컴퓨팅은 곧 모바일 디바이스의 사용시간 단축으로 이어지므로 적합하지 않다. 또한 컨텍스트 상황 추론은 근원지(Source)가 다른 여러 컨텍스트를 고려해야 하는 경우가 있기 때문에 여러 근원지의 컨텍스트를 일괄적으로 관리하는 것이 효과적이다. 그러므로 모바일 컴퓨팅에서의 컨텍스트-상황 추론은 컴퓨팅 파워가 높은 추론 엔진을 통해 수행하는 것이 효과적이다. 즉, 모바일 디바이스는 컨텍스트를 수집해서 추론 엔진에 전송하여 컨텍스트를 저장 및 관리하고 컨텍스트-상황 추론을 추론 엔진에 요청한다. 추론 엔진의 사용으로 모바일 디바이스의 여러 단점을 극복할 수 있다. 하지만 추론 엔진을 설계하기 위해서는 추론 요소들에 대한 이해와 추론 과정에 대한 이해가 필요하다.

이에 본 논문은 3장에서 6가지 컨텍스트-상황 추론 요소로 컨텍스트 측정치, 컨텍스트, 상황, 상황 모드, 상황-컨텍스트 매핑 모델, 추론 함수를 제안하고 4장에서 이들을 실제 사례에 적용하여 제안된 컨텍스트-상황 추론 정형 모델이 실효성 및 범용성을 가짐을 검증한다. 5장에서 추론 요소의 정형 모델을 기반으로 추론 엔진을 설계 및 구현하고 추론 엔진을 이용하여 손/팔 동작 인지를 도메인으로 추론을 실험하여 추론 엔진의 실효성과 재사용성을 검증한다.

2. 관련 연구

Reddy[2]의 연구에서는 모바일 디바이스의 가속도(Accelerometer) 센서를 통해 수집한 3차원 가속도 컨텍스트와 GPS, Wi-Fi, Bluetooth를 이용하여 수집한 위치 컨텍스트로 여러 알고리즘들을 통해 이동 모드(Transportation Mode)를 추론하고 알고리즘간의 정확도를 비교한다. 이동 모드는 단일 사용자의 움직임이 없는 상태, 걷는 상태, 뛰는 상태, 자전거 타는 상태, 차량 탑승 상태를 포함한다. 이 연구에서의 추론 방식은 초기에 각 상태를 나타내는 컨텍스트를 충분히 수집하여 추론에 사용될 모델을 구축한 뒤, 실제 추론 과정에서 초기에 구축한 모델들을 이용하며 추론을 통한 학습은 다루고 있지 않다.

Helaoui[3]의 연구에서는 중간에 개입되고 동시에 발생할 수 있는 활동들을 Markov Logic을 이용하여 인지하는 프레임워크를 제시하고 활동을 인지하기 위한 규칙을 수학적으로 표현한다. 이 연구에서 제안하는 프레임워크는 어떠한 활동의 시작과 끝 시점과 동시에 또는 번갈아가며 수

행하는 활동들을 파악하여 포그라운드(Foreground) 활동과 백그라운드(Background) 활동으로 구분한다. 이 연구는 컨텍스트를 RFID만을 이용하여 수집하였으며 몇 가지 상황에 대한 규칙을 미리 규정하고있다. 즉, 여러 RFID를 지정된 사물 또는 장소에 배치하여 사용자에게 부착한 RFID와 센싱하여 컨텍스트를 수집하기 때문에, 특정 도메인에 의존적이다.

Schmidt[4]의 연구에서는 범용적이고 모듈화가 잘 되어 있는 컨텍스트 모델과 어플리케이션으로 컨텍스트 인지 모바일 미디어 플레이어를 소개한다. 컨텍스트 인지 모바일 미디어 플레이어는 사용자 주변에 있는 디바이스에 미디어를 이주시켜서 사용자가 이동시에도 계속해서 미디어를 볼 수 있게 한다. 이 연구는 컨텍스트 모델링을 위해 Web Ontology Language (OWL)을 사용하고 컨텍스트 탐색을 위해 SPARQL 프로토콜을 사용한다. 이 연구에서는 컨텍스트 수집에 대해서는 상세히 다루고 있지 않으며 컨텍스트 모델을 수학적 보다는 주로 다이어그램을 통해 표현하고 추론은 웹 서비스 형태로 제공하는 컨텍스트 탐색(Query)에 초점을 맞추고 있다.

Ahmed[5]의 연구에서는 의료 청구(Medical Billing)를 도메인으로 의료 보험 청구 거절에 대한 잠재적 에러를 줄이기 위한 규칙 기반 추론 엔진을 설계하였다. 이 추론 엔진은 의료 보험 청구에 있어서 법적 기반 지식과 환자 및 병에 대한 정보를 통해 보험 청구 거절에 대한 잠재적 오류를 추론한다. 규칙 엔진 설계를 위해 이 연구에서는 구조적 탐색 언어(Structured Query Language)를 사용한다. 이 연구는 의료 보험 청구의 도메인에 한정되어 있으며 규칙 기반이기 때문에 도메인 전문가에 대한 의존도가 높다.

Kim[6]의 연구에서는 패턴 인식 기법으로 사람의 행동을 인지하는 모델을 제안한다. 이 연구는 패턴 인식 기법으로 Hidden Markov Model (HMM), Linear Chain Conditional Random Field (CRF), Skip-chain CRF (SSCRF), Emerging Pattern (EP)을 이용하고, 번갈아가며 수행되는 행동에 대한 인지 능력, 모델의 기계 학습 방법, 확장성을 기준으로 이들을 서로 비교하였다. 이 연구는 사람의 행동을 인지하는 도메인에 국한되어 있으며 모델을 수학적인 정형 표현이 아닌 그래프로 표현하고 있다.

기존 연구들의 경우 대부분, (1) 추론 요소들을 식별하고 정의하지 않았고, (2) 컨텍스트 표현이 정형적인 수학 표현 없이 그래프로 표현하고 있으며, (3) 컨텍스트 모델링이 특정 도메인에 한정되어 있고, (4) 추론 역시 특정 도메인에 한정되어 있어서 규칙기반, 패턴 인식 등의 여러 추론 방법 중 특정한 추론 방법에 종속적이며, (5) 추론 수행을 기반으로 모델을 학습시키지 않고 있다. 그러므로 본 연구에서는 컨텍스트-상황 추론 요소를 식별하여 이들에 대한 정형 모델을 제안하고, 컨텍스트-상황 추론 요소를 컴퓨팅 환경에서 실현하기 위한 범용적인 추론 엔진을 설계 및 구현한다. 사용자는 추론 엔진을 통해 컨텍스트 관리, 도메인 관리, 여러 추론 방법을 이용한 추론 수행을 웹 서비스 형태로 제공할 수 있다. 또한, 손/팔 동작 인지를 도메인으로 제안한

추론 엔진을 이용하여 손/팔 동작을 추론하여 추론 엔진의 실효성 및 재사용성을 검증한다.

3. 컨텍스트-상황 추론 요소의 정형명세

모바일 센서 기반의 컨텍스트-상황 추론은 다음과 같은 6가지의 요소인 (1) 모바일 컨텍스트 측정치 (Context Measurement; CM), (2) 모바일 컨텍스트 (Context; CTX), (3) 상황 (Situation; SIT), (4) 상황 모드 (Situation Mode; SMODE), (5) 상황-컨텍스트 매핑 모델 (Situation-Context Mapping Model; SCMM), (6) 추론 함수 (Inference Function; INF)가 존재한다. 각각의 추론 요소들은 Fig. 1 과 같이 센서 기반 컨텍스트-상황 추론의 특정 절차에서 활용된다.

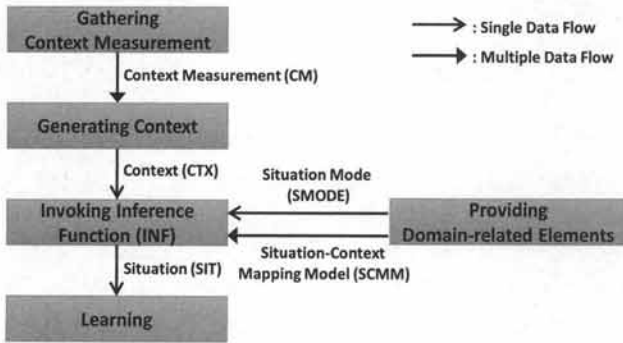


Fig. 1. Process of Context-Situation Inference

모바일 디바이스 센서에서 복수의 컨텍스트 측정치 (CM)를 수집하고 이들을 모바일 디바이스에서 컨텍스트 (CTX)로 그룹화한다. 컨텍스트는 상황-컨텍스트 매핑 모델 (SCMM)을 기반으로 상황 모드 (SMODE)에 포함되어 있는 특정 상황 (SIT)으로 추론 (INF)된다.

컨텍스트 측정치와 컨텍스트는 3.1절에서, 상황과 상황 모드는 3.2절에서, 상황-컨텍스트 매핑 모델은 3.3절에서, 추론 함수는 3.4절에서 자세히 다룬다.

3.1 모바일 컨텍스트 (Context)의 명세

컨텍스트 측정치 (Context Measurement)는 튜플 $CM_{device,cmType}^{time} = (device, cmType, time, cmVal)$ 이다. 이 때, *device*는 컨텍스트 측정치를 수집한 디바이스를, *cmType*는 해당 컨텍스트 측정치의 유형을, *time*은 해당 컨텍스트 측정치가 수집된 시간을, *cmVal*은 수집된 컨텍스트 측정치의 값 벡터를 의미한다.

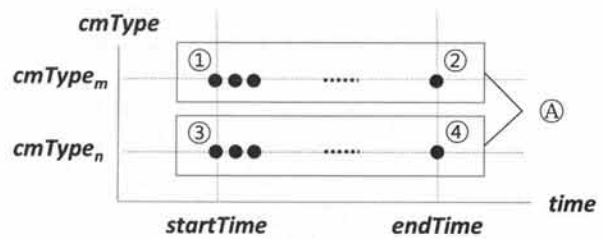
이 때, 컨텍스트 측정치의 값은 컨텍스트 측정치의 유형에 따라 일차원 또는 다차원의 벡터가 된다. 예를들어, 컨텍스트 측정치 유형이 가속도 (Accelerometer)일 때, 컨텍스트 측정치 값 벡터는 3차원이 된다.

컨텍스트 (Context)는 여러 컨텍스트 측정치들을 포함하고 있는 집합이며 다음과 같이 표현된다.

$$CTX_{deviceSet,cmTypeSet}^{startTime,endTime} = \left\{ CM_{device,cmType}^{time} \mid \forall device, cmType, time: device \in deviceSet \wedge cmType \in cmTypeSet \wedge startTime \leq time \leq endTime \right\}$$

이 때, *deviceSet*은 컨텍스트 측정치를 수집한 디바이스의 집합, *cmTypeSet*은 컨텍스트 측정치 유형의 집합, *startTime*와 *endTime*은 각각 컨텍스트 측정치의 시작과 끝 측정 시점이다. 즉, 컨텍스트는 *deviceSet*에 포함된 디바이스가 수집한 컨텍스트 측정치 중에서 *cmType*가 *cmTypeSet*에 포함되고 수집시간이 *startTime*과 *endTime* 사이인 모든 컨텍스트 측정치를 포함하는 집합이다.

컨텍스트는 Fig. 2와 같이 그래프로 표현이 가능하다.



- ①: $CM_{device_i,cmType_m}^{startTime}$ ②: $CM_{device_i,cmType_m}^{endTime}$
- ③: $CM_{device_i,cmType_n}^{startTime}$ ④: $CM_{device_i,cmType_n}^{endTime}$
- Ⓐ: $CTX_{\{device_i\},\{cmType_m,cmType_n\}}^{startTime,endTime}$

Fig. 2. Graph Representation of Context-related Elements

Fig. 2의 ①은 $CM_{device_i,cmType_m}^{startTime}$ 으로 표현 할 수 있으며 컨텍스트 측정치 유형 *cmType_m*의 측정 시점 *startTime*에 측정된 컨텍스트 측정치이다. 이와 유사하게

②는 $CM_{device_i,cmType_m}^{endTime}$, ③은 $CM_{device_i,cmType_n}^{startTime}$, ④는 $CM_{device_i,cmType_n}^{endTime}$ 이다. 사각 박스로 표시되어 있는 영역

Ⓐ는 $CTX_{\{device_i\},\{cmType_m,cmType_n\}}^{startTime,endTime}$ 으로 표현 할 수 있으며 *device_i*에서 시점 *startTime*과 *endTime*사이에서 측정된 컨텍스트 유형 *cmType_m*과 *cmType_n*의 컨텍스트 측정치 컬렉션이다.

컨텍스트의 유형은 컨텍스트가 포함하는 컨텍스트 측정치의 구조에 따라 순차형 (Sequential) 컨텍스트, 관계형 (Relational) 컨텍스트, 복합형 (Complex) 컨텍스트로 구분된다.

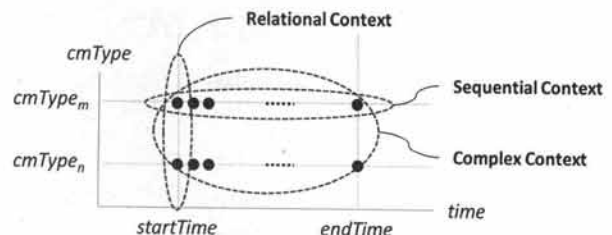


Fig. 3. Graph Representation of Context Types

Fig. 3에서 순차형 컨텍스트는 가로로 열거된, 시간적으로 연속된 컨텍스트 측정치들로 구성된다. 반면에 관계형 컨텍스트는 세로로 열거된, 특정 시점에서 여러 유형의 컨텍스트 측정치들로 구성된다. 복합형 컨텍스트는 시간적으로 연속된 여러 유형의 컨텍스트 측정치들로 구성된다.

순차형 컨텍스트를 예로, 모바일 디바이스 "G3"의 가속도 센서에서 long형식의 초단위 시간 1348486854과 1348488263 사이에 측정된 컨텍스트 측정치들의 값 벡터가 $\langle 7.70185, 3.73901, 3.61174 \rangle$, $\langle -6.69095, -2.99810, -3.50364 \rangle$, ..., $\langle 1.11130, -0.69578, -0.33662 \rangle$ 일 때 다음과 같다.

$$CTX_{(G3),\{Accelerometer\}}^{1348486854,1348488263} = \left\{ \begin{array}{l} (G3,Accelerometer, 1348486854, (7.70185, 3.73901, 3.61174)), \\ (G3,Accelerometer, 1348486855, (-6.69095, -2.99810, -3.50364)), \\ \dots, (G3,Accelerometer, 1348488263, (1.11130, -0.69578, -0.33662)) \end{array} \right\}$$

컨텍스트의 집합을 $ctxSet_i$ 로 간결하게 표현할 수 있다. 이 때, i 는 $ctxSet$ 의 인덱스이다. $ctxSet_i$ 에 포함되어 있는 컨텍스트들을 간결하게 나열하기 위해서 $CTX_{ctxIdx}^{ctxSetIdx} = CTX_{deviceSet,cmTypeSet}^{startTime,endTime}$ 로 표현할 수 있고 CM 은 $cm_{cmIdx}^{ctxIdx} = CM_{device,cmType}^{time}$ 으로 표현할 수 있다. 이 때, $ctxSetIdx$ 는 $ctxSet$ 의 인덱스, $ctxIdx$ 는 ctx 의 인덱스, $cmIdx$ 는 cm 의 인덱스이다. 즉, 어떠한 컨텍스트 집합, $ctxSet_i$ 은 여러 컨텍스트 $ctx_1^i, ctx_2^i, \dots, ctx_n^i$ 을 가지며 한 컨텍스트 ctx_j^i 은 $cm_1^j, cm_2^j, \dots, cm_m^j$ 을 가진다.

3.2 상황 (Situation)의 명세

상황 (Situation)은 실제세계에서 일어나는 사건을 의미하며 튜플 $SIT_{sitName} = (sitName, parent)$ 로 표현된다. 이 때, $sitName$ 은 상황의 이름이며 $parent$ 는 특정 상황의 추상 상황을 의미한다. 알려지지 않은 (Unknown) 상황의 경우 간결화를 위해 sit 으로 표현한다.

Fig. 4와 같이 상황은 $parent$ 인자에 의해 계층적으로 표현될 수 있다.



Fig. 4. Hierarchical Representation of Situations

Level이 높은 상황은 자신보다 낮은 Level의 부모 또는 조상 상황으로 추상화된다. 즉, "Spurting"은 "Running"으로 추상화 가능하고, "Running"은 "Moving"으로 추상화 가능하다.

상황 모드 (Situation Mode)는 상황들의 집합 $SMODE_i = \{sit_1^i, sit_2^i, \dots, sit_n^i\}$ 이다. 이 때, sit_m^i 은 상황 모드 $SMODE_i$ 에 속하는 m 번째 상황을 의미한다. 예를들어, 상황 모드 "Movement"에 상황 "Still", "Walk", "Run"이 속한다면 다음과 같이 표현 할 수 있다.

$$SMODE_{Movement} = \{SIT_{Still}, SIT_{Walk}, SIT_{Run}\}$$

3.3 상황-컨텍스트 매핑 (Situation-Context Mapping)의 명세
컨텍스트로부터 상황을 추론하기 위해서는 어떠한 상황과 그 상황을 나타내는 이미 수집된 컨텍스트들 간의 매핑이 필요하다. 이 매핑을 기반으로 새로 수집한 컨텍스트가 어떠한 상황을 나타내는지 추론한다.

상황-컨텍스트 매핑 모델 (Situation-Context Mapping Model)은 특정 상황이 어떠한 기존 컨텍스트들로 매핑이 되어 있는지를 표현하는 다음과 같은 튜플이다.

$$SCMM_{sitName} = (SIT_{sitName}, ctxSet)$$

이 때, $sitName$ 은 컨텍스트와 매핑할 상황 이름이고 $ctxSet$ 은 상황과 매핑할 컨텍스트 집합이다. 즉, 어떠한 상황은 컨텍스트 집합과 매핑된다. 상황-컨텍스트 매핑 모델은 세 가지 유형으로, 순차형 (Sequential), 관계형 (Relational), 복합형 (Complex)이 존재한다. 각 유형의 결정은 매핑 모델이 다루는 컨텍스트의 유형에 따라 달라진다.

유형 1. 순차형 상황-컨텍스트 매핑 모델 (Sequential SCMM): 순차형 매핑은 순차형 컨텍스트를 상황으로 매핑하는 유형이다. 이 매핑 유형은 특히 패턴 인식에 주로 사용된다. 예를들어, 가속도 센서를 내장한 모바일 디바이스를 어떠한 사람에게 부착하여 그 사람의 걷기, 뛰기 등과 같은 움직임 패턴을 분석할 수 있다. 걷기 상황을 상황-컨텍스트 매핑 모델로 나타내면 다음과 같다.

$$SCMM_{Walk} = (SIT_{Walk}, \{CTX_{(G3),\{Accelerometer\}}^{1348486854,1348488263}, CTX_{(G3),\{Accelerometer\}}^{1348662801,1348662952}, \dots\})$$

이 때, "Walk"라는 상황은 디바이스 "G3"의 가속도 센서에서 수집된 순차형 컨텍스트와 매핑된다.

유형 2. 관계형 상황-컨텍스트 매핑 모델 (Relational SCMM): 관계형 매핑은 여러 컨텍스트들의 관계로부터 상황을 추론하기 위한 유형으로 관계형 컨텍스트를 상황에 매핑한다. 예를들어, 위치가 도로일 때 큰 충격이 발생했다면 이는 사고가 났을 확률이 높다고 추론하는 것과 같다. 이 예에서 사고 상황을 표현하는 상황-컨텍스트 매핑 모델은 다음과 같다.

$$SCMM_{Collision} = (SIT_{Collision}, \{CTX_{(G3),\{Location,Accelerometer\}}^{1348488263}, CTX_{(G3),\{Location,Accelerometer\}}^{1348662801}, \dots\})$$

이 때, 디바이스 “G3”에서 시점 1348488263에서 수집된 Location 측정치와 Accelerometer 측정치를 갖는 튜플과 그 이외의 여러 Location 측정치와 Accelerometer 측정치를 갖는 튜플들이 “Collision”이라는 상황에 매핑된다.

유형 3. 복합형 상황-컨텍스트 매핑 모델 (Complex SCMM): 복합형 매핑은 순차형 컨텍스트와 관계형 컨텍스트의 복합 형태인 복합형 컨텍스트로부터 상황을 추론하기 위한 유형이다. 예를들어, 위치 정보를 이용하여 사용자의 시간별 움직임 속도를 측정하고 가속도 센서를 이용하여 사용자의 움직임 패턴을 분석해서 사용자의 움직임 상태를 알아내는 것과 같다. 복합형 상황-컨텍스트 매핑 모델을 통한 추론은 여러 컨텍스트들의 시간적으로 연속된 정보 및 여러 컨텍스트들 사이의 관계를 통해 상황을 추론하므로 보다 정교한 추론이 가능하다. 사용자의 시간별 위치정보 및 움직임 패턴을 통해 모델링한 “Walk” 상황은 다음과 같이 표현할 수 있다.

$$SCMM_{Walk} = (SIT_{Walk}, \{CTX_{G3, \{Location, Accelerometer\}}^{1348486854, 1348488263}, CTX_{G3, \{Location, Accelerometer\}}^{1348662801, 1348662952}, \dots\})$$

이 때, 디바이스 “G3”에서 수집된 Location 순차형 컨텍스트와 Accelerometer 순차형 컨텍스트가 “Walk” 상황과 매핑된다.

3.4 추론 (Inference) 함수의 정형명세

본 논문에서 제시하고 있는 추론이란 새롭게 발생한 컨텍스트를 기존에 정의된 상황, 기존에 수집된 컨텍스트, 상황과 컨텍스트의 매핑을 통해 비교 및 분석하여 가장 근접한 상황을 도출하는 것이다.

상황-컨텍스트 매핑 유형에 따라 적용될 수 있는 추론 알고리즘이 다르다[8][9][10].

순차형 상황-컨텍스트 매핑 모델일 때: 상황-컨텍스트 매핑 모델이 순차형일 경우 주로 패턴인식에 적합한 Hidden Markov Model (HMM), Conditional Random Field (CRF), K-Nearest Neighbor (KNN), Context-free Grammar (CFG) 등의 알고리즘들이 사용된다. 이 알고리즘들은 기존 사례들과 새롭게 발생한 순차적이고 연속적으로 발생한 컨텍스트들 통해 상황을 추론한다. 새로 발생한 컨텍스트들과 추론된 상황은 이후의 추론 정확도를 높이기 위해 사례로 추가 될 수 있다.

관계형 상황-컨텍스트 매핑 모델일 때: 상황-컨텍스트 매핑 모델이 관계형일 경우에는 규칙 기반 추론 (Rule Based Reasoning; RBR) 알고리즘들이 적용되며 Naïve Bayes (NB), Bayesian Network (BN), Decision Tree (DT), Support Vector Machine (SVM) 등을 포함한다. 이 알고리즘들은 여러 컨텍스트들의 관계에 의해 논리적인 추론이 이루어진다. 이 모델은 새로운 추론에 의해 관계 또는 규칙이 정확도 향상을 위해 학습될 수 있다.

복합형 상황-컨텍스트 매핑 모델일 때: 상황-컨텍스트 매핑 모델이 복합형일 경우, 순차형 모델에서 주로 사용되는 알고리즘과 관계형 모델에서 주로 사용되는 알고리즘들을 복합적으로 사용하여 수행된다. 즉, 복합형 컨텍스트를 순차형과 관계형 컨텍스트로 나누어서 추론을 수행한다. 이 때, 순차형 컨텍스트들을 먼저 추론하고 그들간의 관계를 이후에 추론하는 방법과 관계형 컨텍스트들을 먼저 추론하고 시간적으로 연속된 여러 관계들을 이후에 추론하는 방법이 있다. 이 모델 역시 새로운 추론에 의해 학습될 수 있다.

컨텍스트-상황 추론은 다음과 같이 표현되는 추론함수를 통해 수행된다.

$$INF(alg, smode, mapSet, ctx, n) \rightarrow \{(SIT_i, prob_i) | \forall i \leq n\}$$

이 때, *alg*는 추론에 쓰일 알고리즘, *smode*는 추론으로 도출할 상황의 후보들을 포함하는 상황모드, $mapSet = \{SCMM_{sit} | \forall sit \in smode\}$ 은 추론에 쓰이는 후보 상황들에 대한 상황-컨텍스트 매핑 모델의 집합으로 학습 데이터 선정 기준으로 쓰이며, *ctx*는 상황으로 추론할 컨텍스트로 테스트 데이터이다. 그리고 추론 함수는 추론 결과 상황 SIT과 결과 상황의 확률값 (정확도) *prob*를 포함하는 튜플의 집합을 반환한다. 이 때, 집합의 크기는 추론 함수의 마지막 인자인 *n*이다.

예를들어, 움직임 상황을 추론한다고 할 때, 다음과 같은 추론 함수가 정의될 수 있다.

$$INF(HMM, SMODE_{Movement}, \{SCMM_{Still}, SCMM_{Walk}, SCMM_{Run}\}, ctx, 1)$$

이 때, 도메인으로 상황모드 $SMODE_{Movement}$ 와 상황-컨텍스트 매핑 모델의 집합을 이용하여 추론 알고리즘 HMM으로 새로 발생한 컨텍스트 *ctx*를 $SMODE_{Movement}$ 에 속하는 추론 정확도가 가장 높은 하나의 상황으로 추론한다.

4. 컨텍스트-상황 추론 사례

본 절에서는 3장에서 정의한 추론 요소의 정형명세를 실제 추론 사례에 적용하여 본 논문에서 정의한 정형명세들의 실효성을 검증한다.

이동 모드 추론 사례: 이동 모드 (Transportation Mode)[2]는 사람의 이동 모드에 대한 상황 모드이며 정지 (Still), 걷기 (Walk), 뛰기 (Run), 자전거타기 (Bike), 자동차타기 (Motor) 상태로 이루어져있다. 또한 각각의 상황에 대한 순차형 상황-컨텍스트 매핑 모델이 존재한다. 이 때, 가속도 센서를 내장하고 있는 모바일 디바이스 “G3”를 이용하여 새로 수집한 컨텍스트를 추론 알고리즘 HMM을 통해 이동 모드에 속한 특정 상황으로 추론하고자 한다.

이동 모드 추론 사례는 추론 요소들의 정형명세를 이용하여 Table 1과 같이 표현 할 수 있다.

Table 1. Specification of Transportation Mode Example

Element	Formal Representation of Element
Inference Function	$INF(HMM, SMODE_{Transportation}, mapSet, ctx, 1)$
Situation Mode	$SMODE_{Transportation} = \{SIT_{Still}, SIT_{Walk}, SIT_{Run}, SIT_{Bike}, SIT_{Motor}\}$
A Set of Situation-Context Mapping Models	$mapSet = \{SCMM_{Still}, SCMM_{Walk}, SCMM_{Run}, SCMM_{Bike}, SCMM_{Motor}\}$

추론 함수 인자들을 통해 알고리즘, 도메인, 도메인을 위한 상황-컨텍스트 매핑 모델 집합, 추론할 컨텍스트, 추론 결과 상황 개수를 정한다. 즉, 이동 모드 추론 사례의 추론은 추론 알고리즘으로 HMM을 사용해서 상황모드 $SMODE_{Transportation}$ 에서 추론할 컨텍스트 ctx 에 가장 알맞은 하나의 상황을 추론한다. $SMODE_{Transportation}$ 는 5개의 상황 $SIT_{Still}, SIT_{Walk}, SIT_{Run}, SIT_{Bike}, SIT_{Motor}$ 으로 구성된다. 각각의 상황과 컨텍스트 매핑하기 위해 $SCMM_{Still}, SCMM_{Walk}, SCMM_{Run}, SCMM_{Bike}, SCMM_{Motor}$ 으로 구성되는 $mapSet$ 을 제공해야 한다. 이 추론 사례의 경우, 패턴 인식 추론을 위해 각각의 SCMM은 Hidden Markove Model (HMM)로 재구성된다.

각각의 상황-컨텍스트 매핑 모델들이 저장하고 있는 컨텍스트 집합은 컨텍스트 가시화 도구(Context Visualizer)[7]를 통해 Fig. 5와 같이 차트로 표현된다.

이 때, 새로 발생한 컨텍스트를 각각의 상황-컨텍스트 매핑 모델들이 저장하고 있는 컨텍스트들과 HMM 추론 알고리즘을 이용하여 비교하여 이동 모드에 포함된 상황, Still, Walk, Run, Bike, Motor 중 적합한 상황으로 추론한다. 추론 후, 새로 발생한 컨텍스트를 해당 상황-컨텍스트 매핑 모델에 추가할 수 있다.

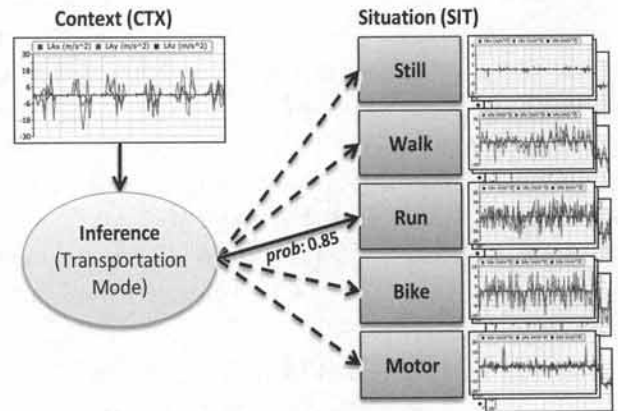


Fig. 6. Inference in Transportation Mode Example

Fig. 6는 새로 발생한 컨텍스트를 이동 모드 (Transportation Mode)의 상황으로 추론 (Inference)했을 때, 추론 결과 상황이 0.85의 확률로 뛰기 (Run)로 결정된 경우를 나타낸다. 즉, Table 1의 추론 함수에서 ctx 가 Fig. 6의 컨텍스트 (CTX)이고 n 이 1일때, 결과 상황과 확률을 포함하는 튜플의 집합 $\{("Run", 0.85)\}$ 을 반환하는 경우이다.

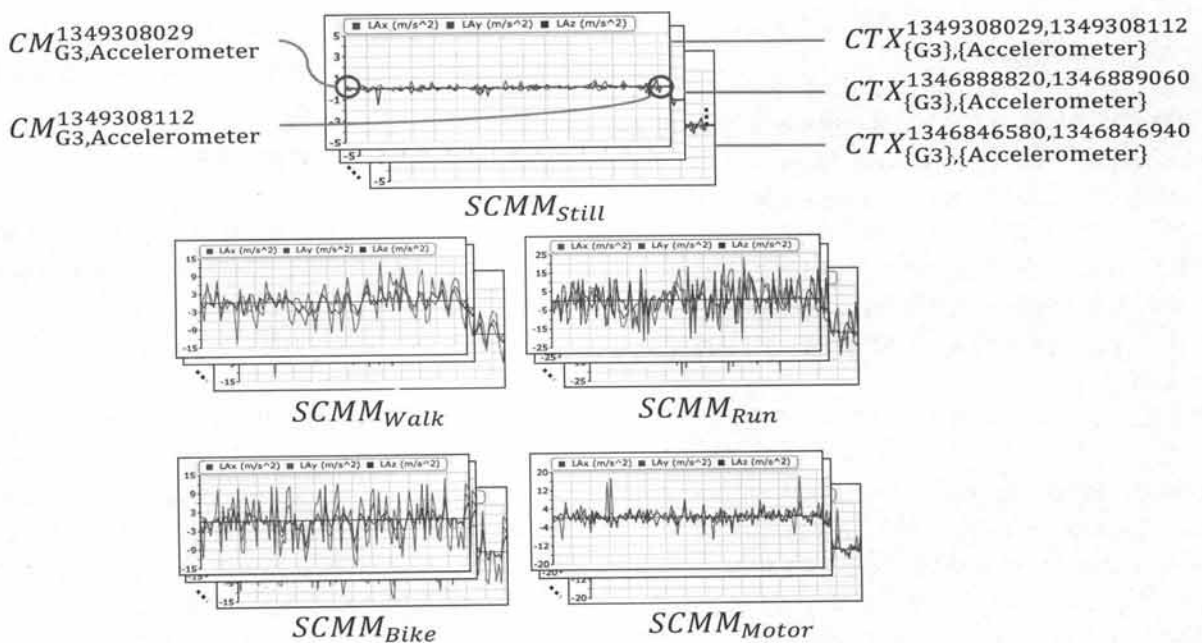


Fig. 5. Context-related Elements Used in Transportation Mode Example

5. 추론 엔진 설계 및 프로토타입 구현

*InfEngine*은 *CTX*, *SIT*, *SMODE*, *SCMM* 등의 추론 요소들을 효율적으로 관리하고 추론 요소들을 기반으로 컨텍스트로부터 상황을 추론할 수 있는 기능을 제공하는 추론 엔진이다. *InfEngine*은 서비스 API를 제공함으로 다양한 분야에서 쉽게 사용할 수 있으며 추론 요소들을 관리하고 컨텍스트를 가시화하는 웹 어플리케이션과 컨텍스트를 수집하여 추론 엔진에 전송하는 안드로이드 기반의 컨텍스트 수집기를 포함한다. 본 논문에서는 프로토타입 *InfEngine*의 설계모델을 기술하고 추론 엔진을 이용한 손/팔 동작 인지 실험을 통해 *InfEngine*을 검증한다.

5.1 *InfEngine*의 아키텍처

*InfEngine*은 모바일 디바이스에서 서버의 기능을 서비스 형태로 사용한다. Fig. 7은 *InfEngine*의 주요 구성 요소와 그들간의 관계를 보여준다.

모바일 디바이스에 설치되는 컨텍스트 모니터 에이전트는 모바일 디바이스에서 컨텍스트를 수집하고 서버로 전송하는 기능을 수행한다. 컨텍스트 모니터 에이전트는 다음과 같은 주요 모듈로 구성된다.

- 컨텍스트 수집기 (CM Gatherer): 컨텍스트 수집기[7]는 모바일 디바이스의 센서들로부터 컨텍스트 측정치를 수집한다. Fig. 8(a)의 컨텍스트 모니터 에이전트 설정 액티비티 (Setting Activity)에서 컨텍스트 측정치를 수집하기 위해 사용할 센서, 컨텍스트 측정치 수집 시간, 컨텍스트

측정치 수집 방식을 설정하여 컨텍스트 측정치 수집 비용을 줄일 수 있다. 컨텍스트 측정치 수집 방식에는 상시 (Continuously) 수집과 주기별 (Periodically) 수집이 있으며 주기별 수집일 경우 사용자가 특정 주기를 결정할 수 있다. 또한을 설정할 수 있다. Fig. 8(b)의 컨텍스트 모니터 에이전트 메인 액티비티 (Main Activity)에서 컨텍스트 수집을 시작/종료할 수 있고 수집 상황을 확인할 수 있다.

- 컨텍스트 빌더 (CTX Builder): 수집한 컨텍스트 측정치를 서버로 전송하기 위해 JSON 등의 형태로 재구성한다. 수집하는 컨텍스트 측정치의 종류 또는 개수가 복수일 경우, 각각의 컨텍스트 측정치 전송을 위해 서버의 API를 호출하는것은 매우 비효율적이다. 그러므로 컨텍스트 필터에서 여러 컨텍스트 측정치를 컨텍스트로 패키지화하는 과정을 수행한다.
- 컨텍스트 전송기 (CTX Transmitter): 컨텍스트 빌더로부터 재구성된 컨텍스트를 넘겨받아 *InfEngine* 서버의 컨텍스트 전송 API를 통해 컨텍스트를 서버로 전송한다. 전송 방식에는 상시 전송과 주기별 전송이 있으며 사용자가 주기별 전송을 위한 주기를 결정할 수 있다. 전송 방식은 Fig. 8(a)의 컨텍스트 모니터 에이전트 설정 액티비티에서 설정할 수 있으며, 주기별 전송의 경우 특정 주기동안 수집된 여러 컨텍스트를 동시에 전송하므로 여러 컨텍스트를 다시 리스트 형태로 재구성한 후 서버로 전송한다. Fig. 8(b)의 컨텍스트 모니터 에이전트 메인 액티비티에서 수집한 컨텍스트 전송을 시작/종료 할 수 있다.

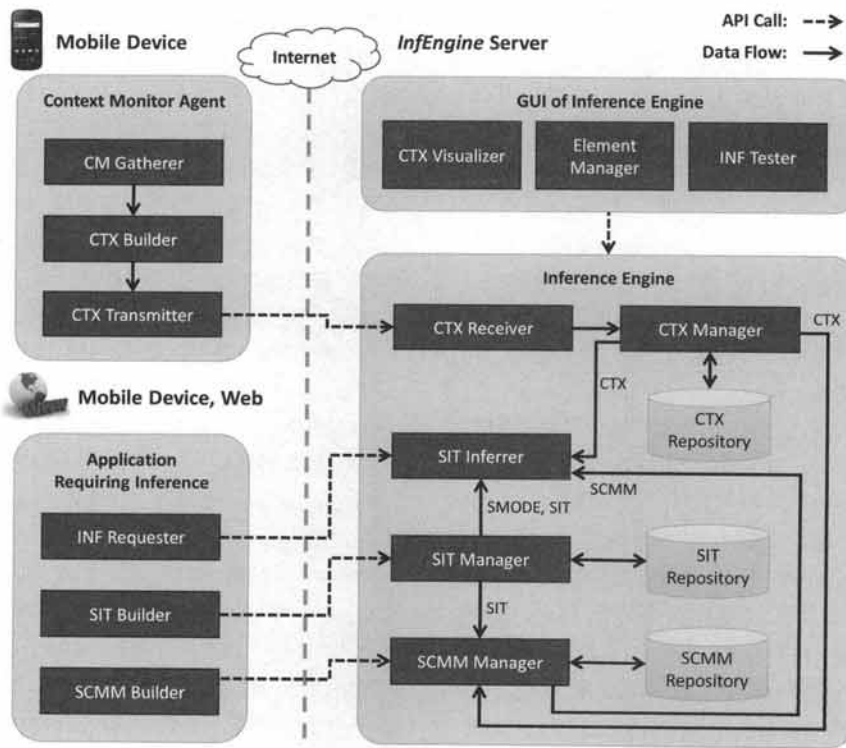


Fig. 7. Architecture of InfEngine

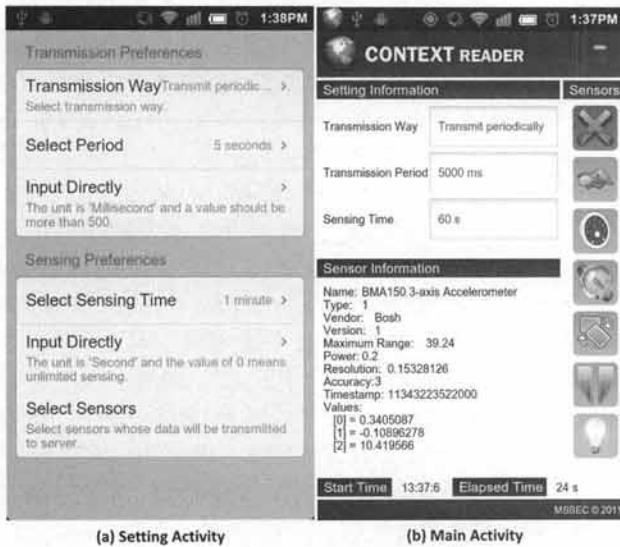


Fig. 8. Context Monitor Agent

추론 요소의 관리와 추론 기능을 제공하는 *InfEngine* 서버의 주요 모듈은 다음과 같다.

- 컨텍스트 리시버 (CTX Receiver): 컨텍스트 모니터 에이전트의 컨텍스트 전송기가 전송한 컨텍스트를 받는다. 전송받은 컨텍스트가 여러 컨텍스트들의 집합일 경우, 이 컨텍스트 집합을 다시 각각의 컨텍스트로 분할하여 컨텍스트 매니저에게 전송한다.
- 컨텍스트 매니저 (CTX Manager): 컨텍스트 리시버가 받은 컨텍스트를 컨텍스트 저장소에 저장하고 저장된 컨텍스트들을 읽어오는 등의 컨텍스트 관리 기능을 수행한다. 컨텍스트 리시버로부터 받은 컨텍스트는 컨텍스트 저장소에 저장되기 이전에 캐시파일에 저장한다. 이는 현재 컨텍스트의 효율적인 제공을 위함으로, 컨텍스트 매니저는 현재 컨텍스트를 컨텍스트 저장소를 거치지 않고 캐시파일을 통해 제공한다.
- 상황 매니저 (SIT Manager): 추론 도메인과 관련된 추론 요소인 상황과 상황모드를 관리한다. 사용자에 의해 상황 모드 (SMODE)와 상황 (SIT)이 추가, 수정, 삭제될 수 있다. 상황모드는 여러 관련된 상황들을 포함하며 한 상황은 여러 상황모드에 속할 수 있다.
- 매핑 모델 매니저 (SCMM Manager): 상황과 컨텍스트간의 매핑 관계를 표현하는 상황-컨텍스트 매핑 모델 (SCMM)을 관리한다. 즉, 사용자는 어떠한 상황을 저장된 여러 컨텍스트와 매핑할 수 있다. 추론 과정에서 이 매핑 모델을 기반으로 학습 데이터를 정한다.
- 상황 추론기 (SIT Inferer): 상황-컨텍스트 매핑 모델을 기반으로 어떠한 컨텍스트를 상황으로 추론한다. 즉, 상황 추론기는 *INF* 함수를 제공하며, *INF* 함수는 사용자가 상황모드, 상황-컨텍스트 매핑 모델의 집합, 추론 알고리즘, 추론할 컨텍스트를 입력하면, 사용자가 입력한 요소들을 통해 테스트 컨텍스트를 상황으로 추론하여 추론 결과를 사용자에게 반환한다.

*InfEngine*은 컨텍스트 관리, 상황 관리, 매핑 모델 관리, 추론 테스트 등의 기능을 GUI로 제공하기 위해 다음 모듈을 포함한다.

- 컨텍스트 가시화 도구 (CTX Visualizer): 컨텍스트 가시화 도구[7]는 숫자, 문자 등의 값으로 이루어져 있는 컨텍스트를 맵, 2D 그래프, 3D 그래프, 도형 등을 이용하여 사람이 쉽게 알아볼 수 있는 형태로 가시화하는 도구이다.
- 추론 요소 관리 도구 (Element Manager): 컨텍스트, 상황, 상황모드, 상황-컨텍스트 매핑 모델 등의 추론 요소들의 관리를 GUI로 제공하는 도구이다.
- 추론 시험 도구 (INF Tester): 추론 함수를 테스트하는 도구이다. 추론 요소 관리 도구에서 추론에 사용될 상황모드, 상황-컨텍스트 매핑 모델, 추론할 컨텍스트를 선택하고 추론 시험 도구에서 추론 알고리즘과 추론 결과 상황의 개수를 입력하여 추론 결과에 대한 보고서를 확인할 수 있다.

추론 엔진과 컨텍스트 모니터 에이전트를 통해 컨텍스트 수집, 추론 요소 관리, 추론 테스트를 수행하여 추론 환경을 갖춘 이후, 추론 어플리케이션에서 실제 추론을 요청하고 추론 결과 상황을 사용할 수 있다. 추론 어플리케이션은 다음과 같은 모듈을 포함한다.

- 추론 함수 호출기 (INF Caller): *InfEngine*의 서버가 제공하는 추론 관련 API를 호출하는 역할을 담당한다. 즉, 추론 함수 호출기는 추론이 필요한 어플리케이션에서 *InfEngine*의 추론 API 호출을 위해 추론 알고리즘, 상황모드, 상황-컨텍스트 매핑 모델 집합, 추론할 컨텍스트 등을 명시해야 한다.
- 상황 빌더 (SIT Builder): 추론 어플리케이션에서 *InfEngine*에 상황을 추가할 수 있다. 상황 빌더가 *InfEngine* 서버의 API를 호출하여 추가할 상황의 명시와 함께 상황 추가를 요청하면 추론 엔진의 상황 매니저가 상황 저장소에 요청된 상황을 추가한다. 이후 추가된 상황은 특정 상황모드에 포함시킬 수 있으며 해당 상황 요소의 매핑 모델을 추가하여 추론에 사용될 수 있다.
- 매핑 모델 빌더 (SCMM Builder): 상황에 대한 상황-컨텍스트 매핑 모델을 추가 또는 수정할 수 있다. 특정 컨텍스트를 매핑 모델에 추가 함으로써 상황을 컨텍스트로 매핑하는 상황-컨텍스트 매핑 모델의 데이터가 풍부해질 수 있으며 특정 상황을 표현하는 적절한 컨텍스트를 매핑 모델에 추가하는 것은 추론의 정확도 증가로 이어진다.

*InfEngine*에는 컨텍스트, 상황, 상황모드, 상황-컨텍스트 매핑 모델 등의 추론 요소를 관리하기 위해 여러 모듈들이 존재한다. 이러한 추론 요소들을 컴퓨팅 환경에서 실현화 (Realize)하기 위해 Fig. 9와 같이 객체 모델을 설계하였다.

추론요소를 다루기 위해 CM, CTX, SIT, SMODE, SCMM 등의 클래스를 도출하였다. 각 클래스는 정형 모델의 추론요소 정형식과 일관성을 유지하였다. 즉, 클래스 CM은 컨텍스트 측정치를, 클래스 CTX는 컨텍스트를, 클래스 SIT은 상황을, 클래스 SMODE는 상황 모드를, 클래스

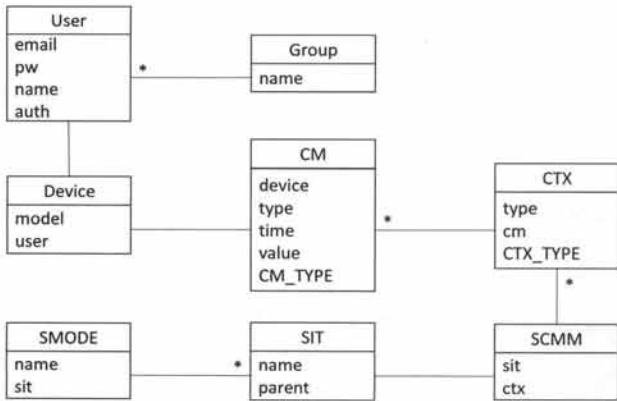


Fig. 9. Object Model of InfEngine

SCMM은 상황-컨텍스트 매핑 모델을 표현한다. 객체 모델에서 볼 수 있듯이, 복수개의 CM이 CTX와 관계가 있으며 복수개의 SIT이 SMODE와 관계를 맺는다. 한 SIT은 SCMM에 의해서 복수개의 CTX로 표현이 된다.

Table 2에서 각 클래스가 포함하고 있는 속성들에 대해 설명한다.

5.2 InfEngine의 프로토타입 구현

InfEngine은 컨텍스트 수집, 관리, 분석에서부터 함께 상황 관리, 추론에 이르는 기능을 제공하는 기능 범위가 넓고 복잡한 시스템이다. 본 논문에서는 InfEngine의 프로토타입을 구현하고 구현물을 기반으로 추론 실험을 수행한다.

프로토타입 InfEngine의 컨텍스트 모니터 에이전트는 안드로이드 기반의 어플리케이션 및 서비스이며 서버는

Python[11] 환경에서 구현되었으며 웹 프레임워크로 Django [12]를 사용하였다. 추론 요소들을 저장할 데이터베이스는 MySQL을 사용하였다. Fig. 9의 객체 모델은 Django의 모델 클래스로 구현되었다. 즉, Django가 제공하는 데이터베이스 동기화 (ORM; 객체 관계 매핑)에 의해 Django 모델 클래스들을 기반으로 데이터베이스 테이블 및 관계 테이블이 생성되고 관리된다.

Table 3에서 프로토타입 InfEngine이 제공하는 API의 URL 패턴, HTTP GET 형식 입력, HTTP Post 형식 입력, 출력에 대해 설명한다.

각각의 API의 URL 패턴과 HTTP GET 형식 입력이 정규표현식 (Regular Expression)으로 표현되어 있다. 이들 표현식 중에서 *ctxId*는 컨텍스트의 ID, *smodeId*는 상황 모드 ID, *scmmId*는 상황-컨텍스트 매핑 모델의 ID, *sitId*는 상황 ID를 의미한다. 즉, *ctxId*의 경우 클래스 CTX와 동기화되어 있는 데이터베이스 테이블에서의 기본키 (Primary Key)를 의미하며 *smodeId*, *scmmId*, *sitId* 역시 이와 유사하게 관련된 테이블의 기본키이다. Fig. 10에서 프로토타입 InfEngine의 GUI를 보여주며 GUI에 포함되어 있는 SMODE List, CTX List, CM List, CTX Chart는 각각 SMODE 리스트 HTML 획득 API, CTX 리스트 HTML 획득 API, CM 리스트 HTML 획득 API, CTX 차트 HTML 획득 API를 사용하여 구성되었다.

프로토타입 InfEngine은 추론을 수행하기 위해 다양한 추론 알고리즘들을 제공하는 Python의 기계 학습 라이브러리인 scikit-learn[13]을 사용한다. 즉, 프로토타입 InfEngine의 상황 추론기에서 추론 요소들을 scikit-learn의 데이터 형식에 맞게 재구성하고 scikit-learn의 API를 호출한다.

Table 2. Description of Class Attributes

Class	Attribute	Description
CM	device	An ID of a device that gathered the context measurement
	type	A type of context measurement (type ∈ CM_TYPE)
	time	Time when the context measurement is gathered by the device
	value	The value of the gathered context measurement ; In the case of acceleration, the form of the value is 3-dimensional vector, <x, y, z>.
	CM_TYPE	A list of context measurement types ; The list includes Location, Acceleration, Orientation, etc.
CTX	type	A type of context (type ∈ CTX_TYPE)
	cm	A set of context measurements
	CTX_TYPE	A list of context types ; The list is composed of Sequential, Relational, and Complex.
SIT	name	The name of the situation
	parent	The parent situation of the situation (An abstract situation)
SMODE	name	The name of the situation mode (Domain name)
	sit	A set of situations included in the situation mode
SCMM	sit	A situation that the situation-context mapping model represents
	ctx	A set of contexts mapped to the situation

Table 3. APIs of Prototype InfEngine

API Name	URL Pattern and GET Input	POST Input (JSON)	Output
Transmitting CMs	^ctxman/send	A set of CMs to be stored	
Browsing CTX	^ctxman/retrieve/(?P<ctxId>\d+)		A set of CMs included in the CTX
Adding SIT	^sitman/add/(?P<smodeId>\d+)	{sitName: SIT name to be added, parentSit: parent SIT name}	
Browsing SMODEs	^sitman/retrieve/(?P<smodeId>\d+)		A set of SITs included in the SMODE
Updating SCMM	^scmmman/update/(?P<scmmId>\d+)	{add: CTX list to be added, remove: CTX list to be removed}	
Inferring SITs	^ctxreasoner/infer/(?P<smodeId>\d+)	Parameters of INF function (CTX for inferring, INF algorithm, etc.)	A set of SITs inferred with probability
Acquiring HTML document for a SMODE List	^render/smodeList\$		HTML document for SMODE list
Acquiring HTML document for a CTX List	^render/ctxList/(?P<sitId>\d+)\$		HTML document for CTX list
Acquiring HTML document for a CM List	^render/cmList/(?P<ctxId>\d+)\$		HTML document for CM list
Acquiring HTML document for a CTX Chart	^render/ctxChart/(?P<ctxId>\d+)\$		HTML document for CTX chart

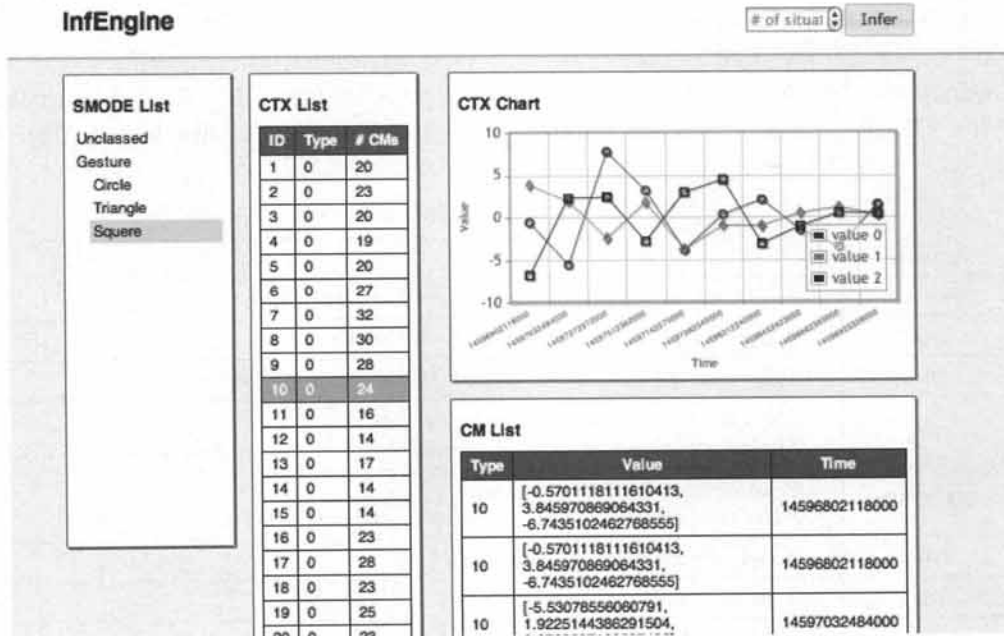


Fig. 10. GUI of Prototype InfEngine

5.3 InfEngine을 이용한 추론 실험

손/팔 동작 인지 (Hand Gesture Recognition): 모바일 디바이스를 잡은 손으로 어떠한 동작을 취하였을 때, 가속도 (Acceleration) 컨텍스트를 기반으로 동작 상황을 추론한다. 손/팔 동작 인지의 추론 요소들은 Table 4와 같다.

손/팔 동작 인지를 위한 추론함수 명세는 이 추론이 HMM을 이용하여 *ctx*를 *SMODE_{HG}*에 포함된 상황 중 하나로 추론한다는 것을 의미한다. 상황-컨텍스트 매핑 모델 집합, *mapSet*에 포함된 여러 컨텍스트들 (Training CTXs)로 구성되는 *SCMM*과 추론함수의 추론할 컨텍스트 (Testing CTX), *ctx*는 Fig. 11과 같다.

Table 4. Specification of Hand Gesture Recognition Example

Element	Formal Representation of Element
Inference Function	$INF(HMM, SMODE_{HG}, mapSet, cts, 1)$
Situation Mode	$SMODE_{HG} = \{SIT_{Circle}, SIT_{Triangle}, SIT_{Square}\}$
A Set of Situation-Context Mapping Models	$mapSet = \{SCMM_{Circle}, SCMM_{Triangle}, SCMM_{Square}\}$

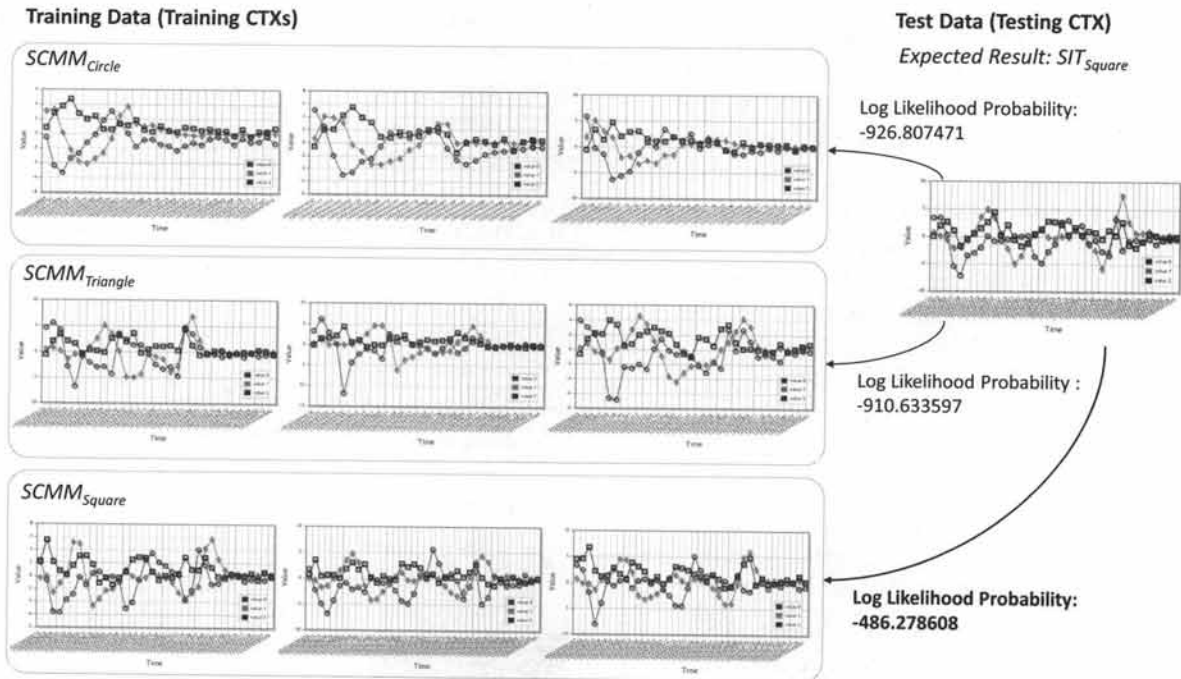


Fig. 11. Experiment of Hand Gesture Recognition Example

SIT_{Square} 가 추론 예상 결과인 테스트 컨텍스트와 세 개의 컨텍스트로 구성된 세 가지 SCMM과의 로그 우도 확률 (Log Likelihood Probability)을 구해 본 결과, SCMM_{Circle}과 -926.807471, SCMM_{Triangle}과는 -910.633597, SCMM_{Square}와는 -486.278608로, SCMM_{Square}와의 확률이 가장 높았다. 즉, 추론 결과로 SIT_{Square} 이 결정되었다.

본 실험의 컨텍스트 수집, 컨텍스트 관리, 상황 관리, 추론 등을 포함하는 추론 프로세스는 프로토타입 *InfEngine*의 API를 사용하여 수행하였다. 본 실험과 마찬가지로, 컨텍스트-상황 추론이 필요한 어플리케이션 개발은 *InfEngine*을 이용하였을 경우, 추론 프로세스에서 필요한 대부분의 기능을 제공받을 수 있기 때문에, 추론을 위한 전체 시스템을 개발할 필요가 없어서 개발 노력을 줄일 수 있다.

6. 결론

컨텍스트 인지 모바일 컴퓨팅은 자동화, 자율화, 맞춤형 서비스를 가능하게 하지만 다양한 컨텍스트 분석, 추론할 상황 분석, 추론에 이용할 알고리즘 분석 등의 복잡한 모델링 과정을 요구한다. 또한 모바일 디바이스의 제한된 자원

으로 이러한 추론 과정을 수행하는 것은 효율적이지 못하다. 그러므로 컨텍스트-상황 추론은 컴퓨팅 파워가 높은 서버에 탑재된 추론 엔진을 통해 수행하는 것이 적합하다.

본 논문에서는 추론 요소와 추론 과정을 보다 명확히 표현하기 위해 (1) 컨텍스트 측정치의 형태가 다른 컨텍스트를 표현하기 위한 추론 요소로 컨텍스트 측정치 (CM)와 컨텍스트 (CTX)를 명세하였고, (2) 컨텍스트 인지를 통해 추론하고자 하는 상황을 표현하기 위한 추론 요소로 상황 (SIT)과 상황 모드 (SMODE)를 명세하였고, (3) 상황과 컨텍스트간의 매핑 관계를 표현하기 위한 모델인 상황-컨텍스트 매핑 모델 (SCMM)을 명세하였고, (4) 컨텍스트-상황 추론을 위한 추론 함수 (INF)를 명세하였다. 또한 컨텍스트-상황 추론 사례에 본 논문에서 제안한 추론 요소의 정형 모델을 적용하여 정형 모델의 실효성이 있음을 보여주었다.

제한한 추론 요소들을 컴퓨팅 환경에서 실현화하기 위해 추론 요소의 정형 모델을 기반으로 추론 엔진을 설계하고 프로토타입 추론 엔진을 구현하였다. 또한 프로토타입 추론 엔진에서 제공하는 기능들을 사용하여 손/팔 동작 추론 실험을 수행하여 추론 엔진 설계의 실효성과 프로토타입 추론 엔진에 대한 실효성 및 재사용성을 확인하였다.

본 논문에서 제안하는 *InfEngine*을 이용함으로써 컨텍스트-상황 추론이 필요한 모바일 어플리케이션, 웹 어플리케이션 등을 개발할 때 컨텍스트 수집, 컨텍스트 관리, 상황 관리, 상황-컨텍스트 매핑 모델 관리, 추론 알고리즘 구현 등을 포함하는 추론 과정을 위한 전체 시스템을 구현할 필요가 없기 때문에 개발 노력을 크게 줄일 수 있다.

참 고 문 헌

[1] Dragan Stojanovic, *Context-Aware Mobile and Ubiquitous Computing for Enhanced Usability: Adaptive Technologies and Applications*, Information Science Reference, 2009

[2] Sasank Reddy, Min Mun, Jeff Burke, Deborah Estrin, Mark Hansen, and Mani Srivastava, "Using Mobile Phones to Determine Transportation Modes," *ACM Transactions on Sensor Networks*, Vol.6, No.2, Article 13, February, 2010.

[3] Rim Helaoui, Mathias Niepert, and Heiner Stuckenschmidt, "Recognizing interleaved and concurrent activities: A statistical-relational approach," *In Proceedings of 2011 IEEE International Conference on Pervasive Computing and Communications (PerCom 2011)*, pp.1-9, 21-25, March, 2011.

[4] Holger Schmidt, Florian Flerlage, and Franz J. Hauck, "A Generic Context Service for Ubiquitous Environments," *In Proceedings of IEEE International Conference on Pervasive Computing and Communications, 2009 (PerCom 2009)*, pp.1-6, 9-13, March, 2009.

[5] Aftab Ahmed, Umair Abdullah and Mohammad J. Sawar, "Software Architecture of a Learning Apprentice System in Medical Billing," *In Proceedings of the World Congress on Engineering (WCE 2010)*, Vol.1, June 20-July 2, 2010, London, U.K.

[6] Eunju Kim, Sumi Helal, and Diane Cook, "Human Activity Recognition and Pattern Discovery," *Pervasive Computing, IEEE*, Vol.9, No.1, pp.48-53, January-March, 2010.

[7] Du Wan Cheun, Moon Kwon Kim, Hyun Jung La, Hyun Joo Bae, Chang Sup Keum, and Soo Dong Kim, "A Practical Framework for Comprehensive Mobile Context Visualization," *In Proceedings of the 8th IEEE International Conference (ICEBE 2011)*, pp.201-206, China, Oct., 2011.

[8] Juan Ye, Simon Dobson, and Susan McKeever, "Situation identification techniques in pervasive computing: A review," *Journal of Pervasive and Mobile Computing*, Vol.8, No.1, pp.36-660, February, 2012.

[9] Sergios Theodoridis and Konstantinos Koutroumbas, *Pattern Recognition*, 4th Ed., Academic Press, 2008.

[10] Ian H. Witten, Eibe Frank, and Mark A. Hall, *Data Mining: Practical Machine Learning Tools and Techniques*, Morgan Kaufmann, 2011.

[11] Python Software Foundation, Python, <http://www.python.org> (accessed by Dec. 14, 2012)

[12] Django Software Foundation, Django, <https://www.djangoproject.com> (accessed by Dec. 14, 2012)

[13] Fabian Pedregosa et al., "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research (JMLR)*, Vol.12, pp.2825-2830, 2011.

[14] Claudio Bettini, Oliver Brdiczka, Karen Henriksen, Jadwiga Indulska, Daniela Nicklas, Anand Ranganathan, Daniele Riboni, "A survey of context modelling and reasoning techniques," *Journal of Pervasive and Mobile Computing*, Vol.6, No.2, pp.161-180, April, 2010.

[15] Anind K. Dey, "Understanding and Using Context," *Journal of Personal and Ubiquitous Computing Archive*, Vol.5, No.1, February, 2001.

[16] Karen Henriksen and Jadwiga Indulska, "Developing context-aware pervasive computing applications: Models and approach," *Pervasive and Mobile Computing*, Vol.2, No.1, pp.37-64, February, 2006.

[17] Daqiang Zhang, Minyi Guo, Jingyu Zhou, Dazhou Kang, and Jiannong Cao, "Context reasoning using extended evidence theory in pervasive computing environments," *Future Generation Computer Systems*, Vol.26, No.2, pp.207-216, February, 2010.

[18] Ahmad Akl and Shahrokh Valaee, "Accelerometer-based gesture recognition via dynamic-time warping, affinity propagation, & compressive sensing," *In Proceedings of IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP 2010)*, pp.2270-2273, 14-19 March, 2010.



김 문 권

e-mail : mkdmkk@gmail.com
 2012년 숭실대학교 컴퓨터학부(공학사)
 2012년~현재 숭실대학교 컴퓨터학과 석사과정
 관심분야: 모바일 컴퓨팅 (Mobile Computing),
 컨텍스트 인지 컴퓨팅 (Context-Aware Computing),
 유비쿼터스 컴퓨팅 (Ubiquitous Computing)



김 수 동

e-mail : sdkim777@gmail.com
 1984년 Northeast Missouri State University 전산학(학사)
 1988년~1991년 The University of Iowa 전산학(석사/박사)
 1991년~1993년 한국통신연구개발단 선임연구원
 1994년~1995년 현대전자 소프트웨어연구소 책임연구원
 1995년 9월~현재 숭실대학교 컴퓨터학부 교수
 관심분야: 서비스 지향 아키텍처(SOA), 클라우드 컴퓨팅(Cloud Computing), 모바일 서비스(Mobile Service), 객체지향 S/W공학, 컴포넌트 기반 개발 (CBD), 소프트웨어 아키텍처(Software Architecture)