

## IoT 기반 상황 인식 추론 시스템 설계 및 구현

이정준<sup>0</sup>, 김경태\*, 송준석\*, 윤희용\*

<sup>0</sup>성균관대학교 정보통신공학과

e-mail:jungjune86@skku.edu<sup>0</sup>, kyungtaekim76@gmail.com\*,  
{alskpo, youn7147}@skku.edu\*

## Design and implementation of context-aware inference system based on Internet of Things

Jung-June Lee<sup>0</sup>, Kyung-Tae Kim\*, Jun-Seok Song\*, Hee-Young Youn\*

<sup>0</sup>Dept. of Electrical and Computer Engineering, Sungkyunkwan University

### ● 요약 ●

IT분야의 하드웨어 발달에 따라 저전력, 저가 및 센싱 및 통신이 가능한 IoT 디바이스들 출시 되고, 이를 이용한 새로운 솔루션 제품들이 출시되면서 IoT 시장 규모가 급격히 확대 되고 있다. 본 논문에서는 이러한 IoT 기기들의 연동을 통한 시너지 효과를 극대화 할 수 있는 상황 인식 추론 시스템의 설계 및 구현에 대해 서술한다. 해당 시스템은 상황 인식을 위한 상황 정보를 기존의 상황 인식 시스템과는 달리 클라이언트 기기에서 규격화 하여 서버로 전송한다. 이러한 이기종 데이터의 규격화는 RDF 문서의 정보 표기 방법인 Triple을 이용해 이루어진다. 이후, 서버로 전송된 상황 정보는 추론 엔진을 통해 현재 상황에 대한 추론 결과를 도출하고, 사용자에게 의해 입력된 룰을 기반으로 상황에 적합한 서비스를 제공한다. 해당 시스템의 정상 동작 확인 여부를 위해, 본 논문에서는 방법 시나리오를 설정하여 테스트를 진행하였다.

**키워드:** 상황인식(Context-aware), IoT(Internet of Things), 추론 시스템(Inference system)

### I. Introduction

IoT 서비스 플랫폼은 사물들 간의 네트워크 연동을 통해 이전의 사람과 사물간의 소통뿐만 아니라, 사물과 사물간의 소통도 가능하게 하며 언제 어디서나 사물들로부터의 데이터 수집 및 제어 방법을 제공한다[1]. 라즈베리 파이, 이두이노 등의 최신 IoT 플랫폼은 이러한 사물간의 접속 개념에서 더 나아가, 메모리, CPU 및 운영 체제를 지원함으로써 단말의 활용 가능성을 더욱 높였다. 이러한 변화는 IoT기기의 기능을 충분히 활용할 수 있는 IoT 기반 시스템에 대한 수요의 증가로 이어지고 있으며, 때문에 본 논문에서는 IoT 기기의 정보 수집 및 통신 기능과 데이터 처리 기능을 살릴 수 있는 IoT 기반 상황 인식 추론 시스템의 설계 및 구현에 대해 서술하고자 한다. 이를 위한 구현 방법으로 이기종 센서 데이터를 활용하기 위해 RDF의 데이터 표현 방식인 3중 구조 형식을 사용한 상황 정보 모델링, 저전력 통신을 위한 CoAP 프로토콜 사용, 경량화 추론 엔진을 위한 전향 추론 기법 및 상황 정보와 룰 데이터의 관리를 위한 MySQL DB를 사용하여 시스템을 구현하였다. 추가로, 사용자 룰을 입력하기 때문에 발생할 수 있는 룰간의 충돌에 대해서는, 가장 최근에 갱신된 상황 정보를 기준으로 룰 실행의 우선순위를 결정한다. 해당 시스템의 테스트를 위해서 방법 시나리오를 기반으로 룰을 생성하여, 이를 이용한 테스트를 실제 장비로 진행하였다.

### II. Preliminaries

#### 1. Related works

##### 1.1 상황 인식 시스템

IoT 분야가 발전함에 따라 상황 인식 시스템을 IoT 분야에서 구현하고자 하는 연구가 진행되고 있다 [1]. 상황 인식 시스템의 핵심 개념은 컴퓨터가 사람처럼 주변 상황 정보를 복합적으로 이해하여 그에 적합한 서비스를 제공하는 것이다. 상황 인식 시스템의 최소 데이터 단위는 상황 정보이다. 상황 정보란 상황을 표현하는 모든 사람, 사물 및 이들 간의 관계 정보를 포함한 모든 것이 될 수 있다. 이러한 추상적인 상황 정보를 표현하기 위해서 반드시 필요한 것은 상황 정보 모델링이다. 상황 정보 모델링을 통해 추상적인 개념의 상황 정보를 현실적이지만 단편적인 데이터를 이용하여 표현이 가능해지며, 이러한 모델링의 규격으로는 Resource Description Framework (RDF) 를 이용한 3중 구조의 표현법과 [2], Ontology를 이용한 OWL 규격 [3]이 널리 사용되고 있다. 본 논문에서는 경량화를 위해 RDF 포맷을 활용한다.

상황 인식 시스템에서 컴퓨터의 상황 인식을 가능케 하는 또 하나의 핵심 기술은 추론 기능에 있다. 상황 인식 시스템에서 추론이란 현재

정보간의 관계를 이용하여 새로운 결론 혹은 결론을 뒷받침 할 수 있는 근거의 도출 및 해당 사항의 진행 과정을 말한다. 이러한 추론을 가능케 하기 위해서는 목적에 맞는 추론 알고리즘과 추론 과정에서 발생할 수 있는 규칙간의 충돌 해결법 [4] 에 대한 정의가 필요하다. 대표적인 추론 알고리즘으로는 전향 추론과 후향 추론이 있다. 전향 추론은 현재 존재하는 상황 정보간의 관계를 이용하여 결론을 도출하는 알고리즘이고, 후향 추론은 현재 상황이 어떠한 과정을 거쳐서 도출되었는지 원인을 찾는 알고리즘이다. 표 1은 두 알고리즘간의 비교에 연구에 의한 분석 결과이다 [5].

Table 1. Comparison between forward chaining and backward chaining

| Attribute       | Forward chaining          | Backward chaining             |
|-----------------|---------------------------|-------------------------------|
| Starts from     | New data                  | Possible conclusion           |
| Aims for        | Any conclusion            | Necessary data                |
| Appropriate for | Planning, monitoring, etc | Diagnostic, prescription, etc |
| flow            | Antecedent to consequent  | Consequent to antecedent      |

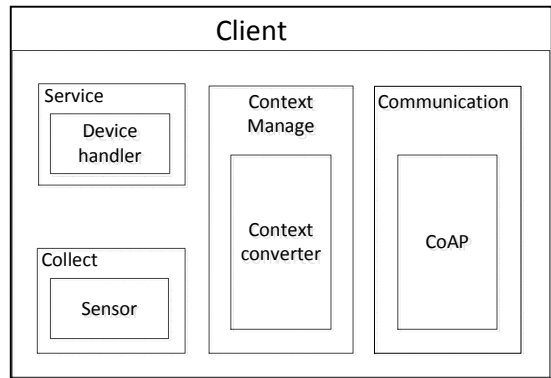


Fig. 2. Architecture of client

그림 2은 데이터 수집 단말의 구조도를 나타낸다. 데이터는 Collect 모듈의 센서를 통해 수집되고, Context Manage를 통해 상황 데이터로 변환한다. 상황 데이터는 3중 구조를 통해 Subject, Predicate, Object 형태로 변하며 이러한 표준화 과정을 통해 이기종 센서 정보의 통합사용이 가능해진다.

### III. The Proposed Scheme

IoT 기반 시스템과 기타 센서 네트워크 시스템의 가장 큰 차이점은 단말 장비의 데이터 처리 능력 및 통신 기능이다. 본 논문에서 제안하는 시스템은 이와 같은 특성을 활용하기 위해 각 센서에서 수집하는 데이터를 수집 단말에서 직접 상황 인식 데이터로 표준화하고, 이를 IoT 단말에 이식된 추론엔진에서 사용자가 입력한 룰을 기반으로 처리하게 된다.

#### 1. 데이터 수집 및 표준화

IoT 단말들은 필요에 따라 다양한 종류의 센서를 사용한다. 이러한 센서에서 수집되는 데이터는 각자 고유한 데이터 포맷을 가지는데, 이를 통합 운용하기 위해서 본 논문에서는 RDF의 데이터 표준화 방법인 삼중 구조를 사용한다. 이러한 3중 구조는 아래의 그림 1과 같이 표현 가능하다.

|                |  |           |        |
|----------------|--|-----------|--------|
| Graphical form | subject  | predicate | object |
| Triple         | subject predicate object   |           |        |
| RDF/XML        | <pre>&lt;rdf:Description rdf:about="subject"&gt;   &lt;ex:predicate&gt;     &lt;rdf:Description rdf:about="object"/&gt;   &lt;/ex:predicate&gt; &lt;/rdf:Description&gt;</pre> |           |        |

Fig. 1. Triple structure for context express

그림 1의 삼중 구조는 RDF포맷에서 데이터를 표현하기 위해 사용하는 구조로써, 영어의 문법과 동일하게 Subject, Predicate, Object의 형태로 상황 정보를 표현 가능하게 함으로써 이 기종 장치간의 데이터 포맷을 상황 정보에 대한 규격화를 가능케한다.

#### 2. 룰 기반 추론 엔진

데이터 수집 단말로 수집한 상황 정보들은 서버 단말로 전송되어 추론 엔진을 이용하여 상황에 적합한 서비스를 제공하는데 사용된다. 이 과정에서 추론 알고리즘은 전향 추론 알고리즘을 사용한다. 또한, 추론 과정에서 발생할 수 있는 충돌에 대해서는, 가장 최신의 데이터를 활용할 수 있는 룰의 우선순위를 최상위로 설정하여, 시스템 실행 중 발생할 수 있는 충돌 상황을 방지하였다.

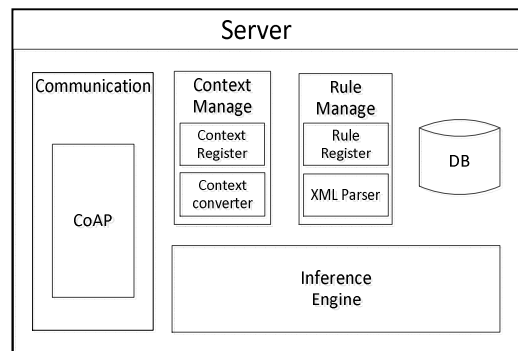


Fig. 3. Architecture of server

그림 3은 룰 기반 추론 단말의 구조도를 나타낸다. 사용자가 특정 상황에 동작하는 서비스를 XML 형태의 룰 문서로 작성하고, 해당 문서는 Rule Manage 모듈에 의해 파싱되어 DB에 저장된다. 이후 데이터 수집 단말을 통해 상황 데이터를 수신 시, 그림 4와 같은 전향 추론 알고리즘을 이용해 추론을 진행한다.

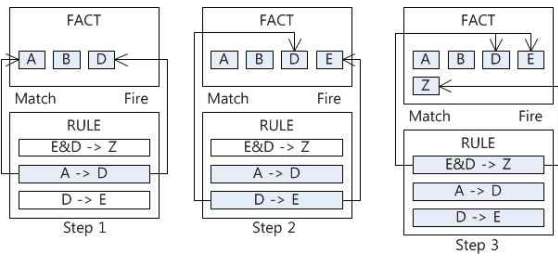


Fig. 4. Forward chaining algorithm

전향 추론과정에서 룰이 매칭 될 경우 룰에 명시된 서비스를 실행하게 되는데, 이 과정에서 다시 현재 상황이 변화하게 되고, 시스템은 이러한 변화를 고려한 추론을 진행한다. 이와 같은 동작은 서버에서 단말로 동작 실행 명령어를 전송하고, 단말에서 실제 동작을 수행함으로써 이루어진다. 클라이언트와 서버의 협업을 통해 이루어지는 전체 서비스의 흐름도는 그림 5 와 같다.

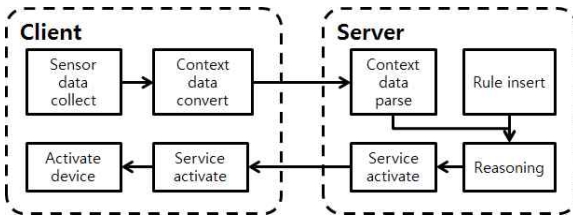


Fig. 5. Service flow of proposed system

#### IV. Scenario Test

본 논문에서 제시한 상황 인식 추론 시스템의 동작을 확인하기 위해, 방법 시나리오를 이용한 테스트를 진행하였다. 해당 시나리오는 집에 사용자 혼자 있는 상황에서, 창문에 움직임이 감지되면 이를 도둑이 침입했다고 인식하고, 알람 경보를 울린 후 가까운 경찰서에 연락한다. 이를 테스트하기 위한 하드웨어 구성으로는 Rspberry Pi B+ 3식과 HD-SEN0018 모션센서, COM05051P 스피커를 사용하였으며, 소프트웨어로는 Raspbian OS, Mysql DB 및 Callifonium CoAP 라이브러리[6]와 자체 제작한 전향 추론 엔진을 사용하였다. 표 2는 이러한 시나리오 테스트를 위해 설정한 룰 정보를 나타낸다.

Table 2. Rules of security system

| Rule # | Precondition       | Postcondition |
|--------|--------------------|---------------|
| 1      | window movement    | alarm on      |
| 2      | living_room empty  | thief enter   |
|        | alarm on           |               |
| 3      | thief enter        | call police   |
|        | house status alone |               |

이와 같이 룰을 설정한 상태에서, 모션 센서를 창가에 배치한 후, 창가에서 움직임을 통해 모션 센서에 입력을 주고, 이후 상황 정보 발생 및 추론 엔진에 의한 경찰 호출 시나리오가 정상 작동함을

확인하였다. 표 3은 해당 과정에서 발생한 상황 정보이다.

Table 3. Context data of security system

| Rule #  | Precondition       |
|---------|--------------------|
| Default | house status alone |
|         | living room empty  |
|         | alarm off          |
| 1       | window movement    |
| 2       | alarm on           |
| 3       | thief enter        |
| 4       | police call        |

실험 과정에서 최초 상황 정보는 Default의 정보로, 집안에 혼자 있는 상태를 의미한다. 이후 창문의 모션 센서를 통해 Context #1의 정보가 입력되면, Rule #1과 매칭이 되어 alarm on 명령어가 실행된다. 이는 알람 발생 및 Context #2를 생성하며, 이후 순차적으로 Context #3, #4 또한 생성되어 최종적으로 도둑 침입 시 경찰을 자동으로 호출하는 서비스를 제공하게 된다.

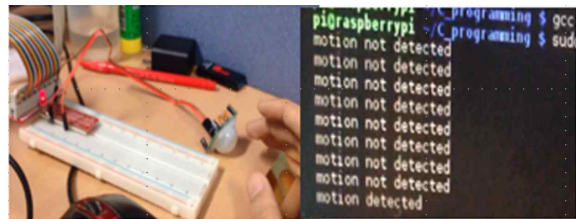


Fig. 6. Motion detection by using sensor

#### V. Conclusions

본 논문에서는 IoT 단말의 기능을 극대화 하기 위한 IoT 기반 상황 인식 추론 시스템의 설계 및 구현에 대해 서술하였다. 이기종 센서의 데이터를 통합 처리 하기 위해 3중 구조를 이용하여 데이터를 상황 데이터로 표준화 하였고, 이를 전향 추론 기법을 이용한 추론엔진을 통해 사용자에게 적합한 서비스를 제공하도록 하였다. 추후 연구로는 사용자 편의를 위한 룰 생성 도구 및 서드 파티와의 연동을 위한 상황 정보 모델링에 대해 보다 깊은 연구를 진행하고자 한다.

#### Acknowledgment

본 연구는 BK21Plus 사업, 한국연구재단 기초연구사업 (2013R1A1A2060398), 삼성전자, 미래창조과학부 및 정보통신기술 연구진흥센터의 정보통신·방송 연구개발사업 (1391105003)의 일환으로 수행하였음.

## References

- [1] Charith Perera, et al., "Context Aware Computing for the Internet of Things: A Survey," Communications Surveys & Tutorials, IEEE, Vol. 16, Issue 1, pp. 414-454, July 2014.
- [2] w3.org, "Resource description framework(rdf)," 2004, <http://www.w2.org/RDF/>
- [3] w3.org, "Web ontology language(owl)," Webpage, October 2007, <http://www.w2.org/2004/OWL/>
- [4] Wang Yuan, et al., "Design and Implementation of Inference Engine for Conflict Resolution," Intelligent System Design and Engineering Application (ISDEA), 2012 Second International, pp. 220-223, Jan 2012.
- [5] Al-Ajlan, "The Comparison between Forward and Backward Chaining," International Journal of Machine Learning and Computing, pp 106-113, Apr 2015.
- [6] M. Kovatsch et al., "Californium: Scalable cloud services for the Internet of Things with CoAP," Internet of Things (IOT), 2014 International Conference on the, pp 1-6, October 2014.