

개인 프레즌스-선호 기반 지능형 로컬 서비스 시스템과 모바일 엣지 컴퓨팅 환경에서의 적용 방안

김길환[◦], 장진산^{*}, 금창섭^{**}, 정기숙^{**}

Individual Presence-and-Preference-Based Local Intelligent Service System and Mobile Edge Computing

Kilhwan Kim[◦], Jin-San Jang^{*}, Changsup Keum^{**}, Ki-Sook Chung^{**}

요 약

IoT 환경이 심화됨에 따라 집, 사무실 등 특정 지역에 설치된 센서 정보를 활용하여 지역내 냉난방, 조명 등의 서비스를 자동 조절하는 지능형 로컬 서비스에 대한 관심이 커지고 있다. 그런데 지금까지의 IoT 기반 지능형 로컬 서비스는 지역 내 사용자의 프레즌스와 서비스 선호도를 간접적인 방식으로 반영함으로써 실제 체감 중인 사용자의 선호도를 왜곡하여 반영하는 문제가 발생한다. 본 연구에서는 이러한 문제점을 해결하기 위해 개별 사용자의 프레즌스 및 선호도 정보를 기반으로 한 지능형 로컬 서비스 제어 방식을 제안하고 이를 프로토타입으로 구현한 결과를 제시한다. 아울러 대부분의 지능형 로컬 서비스를 위한 복잡한 예측 모형의 생성은 주로 클라우드 상의 서버에서 수행되어 왔다. 그러나 이러한 방식은 IoT 기기와 클라우드 간의 대량의 데이터 전송을 발생시킨다. 모바일 엣지 컴퓨팅 환경은 지능형 로컬 서비스 제어 시스템의 이러한 문제점을 해결할 수 있는 해결책이 될 수 있다. 본 연구에서는 클라우드 환경에서 개인 프레즌스-선호 기반 지능형 로컬 서비스 시스템을 구현한 후, 구현 결과를 기반으로 모바일 엣지 컴퓨팅 환경에 적용하는 방안을 제시한다.

Key Words : intelligent service, Internet-of-Things (IoT), data mining, presence information, edge computing

ABSTRACT

Local intelligent services aim at controlling local services such as cooling or lightening services in a certain local area, using Internet-of-Things (IoT) sensor data in the area. As the IoT paradigm has evolved, local intelligent services have gained increasing attention. However, most of the local intelligent service mechanism proposed so far do not directly take the users' presence and service preference information into account for controlling local services. This study proposes an individual presence-and-preference-based local service system (IPP-LISS). We present a intelligent service control algorithm and implement a prototype system of IPP-LISS. Typically, the intelligence part of IPP-LISS including the prediction models, is generated on remote server in the cloud because of their compute-intense aspect. However, this can cause huge data traffic between IoT devices and servers in the cloud. The emerging mobile edge computing technology will be a promising solution of this challenge of IPP-LISS. In this paper, we implement IPP-LISS in the cloud, and then, based on the implementation result, we discuss applying the mobile edge computing technology to the IPP-LISS application.

※ 이 논문은 정부의 재원으로 한국전자통신연구원의 주요사업으로 수행된 연구임 (17ZH1510, 단말 근접 실시간 스마트 서비스추진 플랫폼 기술 개발)

◦ First and Corresponding Author : Sangmyung University Department of Management Engineering, khkim@smu.ac.kr, 정희원

* Sangmyung University Department of Management Engineering, jinsan26@naver.com

** ETRI Hyper-connected Communication Research Laboratory, cskeum@etri.re.kr, 정희원, kschung@etri.re.kr, 정희원

논문번호 : KICS2016-12-372, Received December 4, 2016; Revised January 10, 2017; Accepted January 16, 2017

I. 서 론

네트워크 자원의 이용률 증가는 이동통신사업자에 게 큰 도전 요소이다. 네트워크 자원의 이용률 증가의 주요 원인 중 하나는 모바일 및 IoT 기기에 의한 데이터 트래픽의 증가이다. 대다수의 모바일 및 IoT 응용이 원격의 데이터 센터에 있는 서버에 의존하고 있어 이러한 모바일 및 IoT 기기와 인터넷에 연결된 데이터 센터 사이에는 많은 데이터의 트래픽이 발생한다. 향후 새로운 모바일 및 IoT 응용의 등장은 이러한 추세를 더 심화시킬 것이다. 특히 웨어러블 기기와 IoT 기기의 컴퓨팅 능력이 향상될 것이므로 이에 따른 새로운 응용 서비스의 도입과 함께 데이터 통신에 대한 요구량도 폭발적으로 늘어날 것으로 예상된다.

늘어나는 통신 요구량에 맞추어 이동통신사업자는 모바일 엣지에서 LTE Advanced 등 신기술을 적용하고 있고, 이러한 신기술은 모바일 엣지에서 더 넓은 대역폭을 제공하고 있다. 그러나 모바일 엣지에서 커진 대역폭만큼 망 중심의 부하를 증가시키고 더 많은 투자로 귀결될 수밖에 없다.

이러한 문제를 해결하기 위해 등장한 개념이 모바일 엣지 컴퓨팅(Mobile Edge Computing, 이하 MEC) 기술이다.^[1-3] 전통적으로 기지국은 데이터 트래픽을 전송만 하지 이를 분석하거나 사용자의 요청에 반응하지는 않는다. 그러나 MEC 환경에서는 모바일 엣지에 MEC 서버라는 컴퓨팅 및 저장 장치를 도입한다. 이를 통해 MEC 서버는 자신이 처리할 수 있는 트래픽은 스스로 처리하여 응답하고, 그렇지 않은 서비스는 원래의 서버로 전송함으로써 망 중심의 트래픽을 감소시키고, 사용자에게는 빠른 응답시간을 확보한다. 또한 응용 서비스 제공자(Application Service Provider, 이하 ASP) 입장에서는 데이터 센터의 컴퓨팅 자원에 대한 투자를 경감시킬 수 있는 이점이 있다.

한편 IoT 환경이 심화됨에 따라 집, 사무실 등 특정 지역내에 센서 정보를 활용하여 지역내 냉난방, 조명 등의 서비스를 자동 조절하는 지능형 로컬 서비스 제어에 대한 관심이 점차 커지고 있다.^[4,5] 그런데 지금까지의 IoT 기반 지능형 로컬 서비스는 다음의 두 가지 문제점을 가지고 있었다.

첫째, 사무실 등 특정 지역내의 사용자의 프레즌스 와 서비스 선호도를 간접적인 방식으로 수집하여 서비스 자동 조절에 반영함으로써 현재 재실중인 사용자의 실제 선호도를 왜곡하여 반영하는 문제가 발생한다. 예를 들어 냉방의 경우 대부분의 방식이 현재의 온습도 등의 열환경 센서 정보를 이용하여 지역내의

사용자가 표준적인 사용자 선호를 가질 것이라는 가정 하에 냉난방을 조절하거나, 또는 지역내 냉방기의 조정 이력에 의해 사용자의 간접적 선호도를 파악하여 냉난방을 자동 조절을 하고 있다.^[6-9] 그러나 이러한 방식은 지역내 사용자가 여러 명이고 이들의 지역내 프레즌스가 시시각각 변할 때, 사용자의 프레즌스 여부에 따라 시시각각 바뀌는 지역내 사용자의 선호도를 반영할 수 없다는 단점이 있다. 예를 들어 현재 지역내에 약한 난방을 원하는 사용자가 대다수인데 지금까지 냉방기 조절을 주로 수행했던 사용자가 강한 난방을 선호하였다면, 지능형 로컬 서비스 제어 알고리즘은 강한 난방 선호에 대해 학습되어 있으므로 현재 재실중인 사용자의 선호를 반영하지 못한 채 쓸데없이 에너지를 낭비하면서 사용자의 만족도는 높이지 못할 것이다. 이러한 경우 개별 사용자의 프레즌스 및 선호도 정보를 결합하여 각 시점에 맞는 지능형 로컬 서비스를 수행한다면 에너지 효율과 함께 사용자 만족도를 더 한 층 향상시킬 수 있을 것이다.

둘째, 기존의 지능형 로컬 서비스 제어의 아키텍처는 서비스 자동 조절 모형의 생성과 관리가 클라우드 환경에서 수행되는 것을 명시적으로 또는 암시적으로 상정하고 있다.^[6-9] 이는 지능형 서비스를 위해서는 데이터 마이닝 등의 데이터 분석 기법을 통해 사용자의 선호도를 학습할 필요가 있는데 이는 매우 계산 집약적인 작업이어서 IoT 기기 등에서 수행할 수 없기 때문이다. 그래서 Amazon EC2나 Windows Azure 같은 원격 클라우드 상의 서버에 사용자의 선호도를 학습하는 작업을 맡기고, IoT 기기는 각종 데이터를 클라우드 상의 서버에 전달하는 역할만 수행한다. 그러나 이러한 방식은 IoT 기기와 클라우드 간의 대량의 데이터 전송을 야기시킬뿐 아니라 실행 시 긴 지연 시간이 발생할 수 있다. 이러한 문제를 해결하기 위해 원격의 클라우드 상의 서버가 아니라 지역내의 Wi-Fi AP를 이용하여 계산을 수행하는 방법이 제안되기도 하였다.^[10] 그러나 Wi-Fi AP도 낮은 컴퓨팅 자원을 가지고 있으므로, 응용에 대한 수요가 늘어나면 규모 확장이 가능하지 않은 방법이라 하겠다. 또한 지역내에 Wi-Fi 접속이 가능한 지역도 제한이 있어 응용의 전개에 한계를 가지고 있는 문제점이 있다.

현재의 지능형 로컬 서비스 제어 시스템의 이러한 한계를 넘어서는데 MEC 환경은 매우 훌륭한 대안이 되리라 예상된다. 기존의 클라우드 상의 서버가 아니라 모바일 엣지 상의 MEC 서버에서 지능형 서비스를 위한 예측 모형을 생성한다면 망 중심을 통해 원격의 클라우드까지 대량의 IoT 데이터를 전송할 필요성이

감소하고, 모형 생성 및 관리에 안정적인 반응 시간의 확보가 예상된다. 또한 사용자의 프레즌스 정보 등을 IoT 기기뿐 아니라 기지국에 연결되어 있는 MEC 서버의 정보를 함께 이용할 수 있으므로써 사용자 프레즌스-선호 기반 지능형 로컬 서비스의 구현이 더 한층 용이해 질 것이다.

본 연구는 개인 프레즌스-선호 기반 지능형 로컬 서비스 시스템(Individual Presence-and-Preference-based Local Intelligent Service System, 이하 IPP-LISS)의 서비스 시나리오와 아키텍처를 제시하고, 냉난방 서비스 제어를 위한 서비스 제어 알고리즘을 제시한다. 아울러 클라우드 환경 하에서 IPP-LISS의 프로토타입 시스템을 구현한다. 이러한 구현을 통해 얻은 분석 결과를 이용하여 MEC 환경에서 이 시스템을 어떻게 적용하여야 하는지를 논의한다.

이를 통해 본 연구가 기여하고자 하는 바는 다음과 같다. 첫째, 기존의 표준적인 사용자 선호도 학습 중심의 지능형 로컬 서비스 제어 방식을 대체할 수 있는 사용자 프레즌스-선호 기반 지능형 로컬 서비스 제어 방식을 새롭게 제안함으로써 지역내 사용자의 만족도는 더욱 향상시키고 에너지 사용을 절감할 수 있는 방안을 제시한다. 이러한 로컬 서비스 제어 방식은 냉난방 서비스뿐 아니라, 조명, 음향 등의 다양한 지역 서비스를 자동 조절하는 새로운 방식으로 활용될 수 있을 것이다. 둘째, 기존 MEC에 대한 연구는 일반적인 플랫폼 중심의 논의였지^[1-3], 구체적인 서비스 시나리오와 구현 방식에 대한 제시는 적었다. 본 연구는 지능형 로컬 서비스 제어 시스템이 MEC 환경에서 어떻게 구현될 수 있는지 제시함으로써 MEC 플랫폼에 대한 연구에 있어 구체적으로 활용할 수 있는 유스 케이스로서의 기여가 있다 하겠다.

본 논문의 구성은 다음과 같다. 2장에서 MEC와 IoT 기반 지능형 로컬 서비스 제어와 관련된 기존 연구를 살펴본다. 3장에서 IPP-LISS 시스템이 상정하는 서비스 시나리오와 전반적인 시스템 아키텍처에 대하여 논의한다. 다음으로 4장과 5장에서 개인 프레즌스-선호 기반 지능형 로컬 서비스 제어 알고리즘을 설명하고 이를 실제적으로 어떻게 구현하였는지를 논한다. 6장에서 MEC 환경에서 IPP-LISS의 구현 방안을 제시하고, 마지막으로 연구의 시사점 및 향후 연구 방향을 제시하도록 한다.

II. 문헌 연구

2.1 MEC 연구 동향

MEC는 아직 유아기 단계의 연구 주제로서 부분적인 연구와 구현은 있었지만 이를 포괄적으로 구현한 결과는 아직 없다고 할 수 있다.^[1,3] 대표적인 관련 연구는 다음과 같다. FemtoClouds^[11]는 Cloudlet^[10]처럼 모바일 클라우드 컴퓨팅(Mobile Cloud Computing, 이하 MCC) 개념을 구현한 플랫폼이다. 모바일 기기의 계산 집약적인 작업을 클라우드로 보내지 않고 모바일 클라우드 안에서 오프로딩(offloading)하여 해결하는 시스템으로서, 모바일 엣지 컴퓨팅이라기보다는 모바일 클라우드 컴퓨팅 관련된 연구라 할 수 있다. REPLISOM^[12]은 LTE의 evolved NodeB (eNB)에 클라우드 기능을 추가하는 방식에 대한 연구로서 개별 IoT 기기가 자신의 메모리 객체를 밀어(push)내어 전달하는 것이 아니라, 장치 대 장치 통신을 통해 메모리 객체를 합쳐서 압축한 후 MEC 서버가 당기(pull)는 방식으로 모바일 기기의 작업을 오프로딩하는 플랫폼을 제안하였다. ME-VoLTE^[13]는 Voice over LTE가 가능한 MEC 구조를 제시하여 모바일 기기는 인코딩 작업 등을 MEC 서버에 오프로딩함으로써 배터리 소모를 감소시킬 수 있도록 하였다. Chen et al.^[14]은 분산형 작업 오프로딩 모형을 제시하고 게임 이론을 기반으로 내쉬(Nash) 균형을 추구하는 알고리즘을 제시하였다. 살펴본 바와 같이 MEC에 대한 연구는 플랫폼의 구현 방식을 중심으로 이루어져 있고 구체적인 응용 서비스의 시나리오의 제시는 많지 않다. 간혹 응용 서비스 시나리오가 제시되기는 하지만 주로 모바일 클라우드 컴퓨팅 측면이고 모바일 엣지 컴퓨팅과 관련된 구체적인 응용 시나리오와 구현 방안에 대한 연구는 적다.

Beck et al.^[1]을 따르면 MEC 환경에서 적합한 응용 서비스 분야는 (1) 작업 오프로딩, (2) 엣지 콘텐츠 전달, (3) 데이터 병합(Aggregation) (4) 지역내 연결, (5) 콘텐츠 스케일링, (6) 정보부가(Augmentation) 응용으로 나눌 수 있다.

본 연구는 6장에서 살펴볼겠지만 MEC 환경에서 작업 오프로딩 방식이 적용될 수 있는 응용 시나리오라고 할 수 있다. 특히 계산 집약적인 예측 모형 생성을 MEC 서버에 위탁하는 시나리오이다. 앞서 MEC 연구 동향에서 살펴 본 바와 같이 모바일 엣지 컴퓨팅 기술이 적용되는 응용 시나리오에 대한 연구가 적으므로 본 연구는 이러한 부분에 기여가 있다.

2.2 지능형 로컬 서비스 제어 연구 동향

IoT 환경이 급속히 발전함에 따라 특정 지역에 대한 센서 정보를 이용하여 지역내의 상황을 인지하여 적절한 방식으로 지역내의 냉난방, 조명, 환기, 방범 등의 서비스를 자동 조절하려는 시도가 활발해지고 있다. 서비스를 자동 조절하는 방식으로는 지역내의 센서 정보와 데이터 마이닝 기법 등을 이용하여 사용자의 선호나 지역내의 상황을 적절히 예측하는 모형을 학습시킨 후, 이를 서비스 제어 알고리즘에 사용하는 방식이 많이 사용된다.^[4] 본 연구에서는 이러한 지역내의 사용자의 선호나 상황 인지를 학습의 과정을 통해 모형을 생성한 후 서비스를 자동 조절하는 메커니즘을 통칭하여 지능형 로컬 서비스 제어라고 부르도록 한다.

대표적인 지능형 로컬 서비스 제어 분야가 스마트 홈^[5,7,15,16]과 스마트빌딩^[6,9,17]에서의 냉난방, 조명, 방범 서비스 등이라 할 수 있다.

그런데 이러한 지능형 로컬 서비스 제어 시스템에 대한 연구는 대부분 지역내의 사용자 프레즌스 정보를 고려하지 않고 지역내의 표준적인 사용자를 상정하여 서비스 선호나 상황 인지 모형을 훈련시킨 후 이를 바탕으로 서비스를 조절하거나^[6,7], 사용자의 프레즌스를 파악하여 모형을 학습하더라도 개별 사용자의 선호를 직접적으로 파악하여 반영하는 메커니즘을 가지고 있지 않다.^[13,15] 반면 본 연구는 지역내의 개별 사용자의 프레즌스와 선호를 직접적으로 파악하여 모형을 학습시킨 후, 사용자의 프레즌스의 변화에 따라 동태적으로 서비스를 조절해 나간다는 데 그 차이를 가진다.

아울러 기존의 연구에서 지능형 로컬 서비스 제어 시스템에서 자동 조절을 위한 모형을 학습하는 부분은 많은 컴퓨팅 요구량이 필요하므로 원격의 클라우드에서 수행되는 것을 명시적으로나 암시적으로 상정하였다. 그러나 IoT 환경이 심화됨에 따라 원격의 클라우드에서 학습을 처리하는 것은 대량의 데이터의 전송을 야기하여 전체 통신 시스템 하에서 최적화된 방법이 아니라 할 수 있다. 또한 대규모 빌딩뿐 아니라 소규모 사무실, 카페, 공연장, 집 등 다양한 지역에서 지능형 로컬 서비스를 하기 위해서 모든 지역에 별도의 서버를 설치할 것은 과도한 요구사항이 될 것이다. 따라서 모바일 엣지에서 모형을 학습하는 방안에 대한 연구가 필요하다.

그런데 지금까지 모바일 엣지에서 IoT 관련 연구는 주로 기기 대 기기 방식과 압축 샘플링 방식을 이용하여 IoT 기기의 LTE 상에서의 메모리 복사 구조 및 프

로토콜 제시하거나^[12], 소형셀 환경에서 사용자 컨텍스트 기반 무선 캐시 알고리즘을 제시하거나 브로드캐스트 방법^[18,19] 등 프로토콜 및 플랫폼 구현 방안에 대한 논의로 제안되어 있다. IoT 상에서 모바일 엣지 컴퓨팅을 적용하는 구체적인 응용 시나리오의 제시는 거의 없다. 반면 본 연구에서는 지능형 로컬 서비스 제어 시스템을 구현하는데 그치지 않고 모바일 엣지 컴퓨팅 환경에 이를 적용하는 개념적 방안을 제시하여 IoT 환경에서의 모바일 엣지 컴퓨팅을 적용하는 응용 시나리오를 제시하였다는데 그 차이점을 가진다. 6장에서 보듯이 IoT 환경에서 데이터 처리 시간이 길고, 저장 공간이 많이 필요하고, 지연에 대해 민감하지 않은 작업에 MEC 적용의 필요성이 많음을 볼 수 있다.

III. IPP-LISS의 서비스 시나리오 및 아키텍처

집, 사무실, 지하철, 버스, 공연장, 전시장, 회의실, 교육장, 카페 등의 지역공간에 머무는 다수의 개인들에게 무차별적으로 동일하게 제공되는 냉난방 서비스, 음악 서비스 등의 로컬 서비스에 대한 제어는 대부분 그림 1 같이 중앙집중형으로 수행되어 왔다. 그 지역공간에 대한 배타적 권한을 갖고 있는 관리자나 로컬 서비스의 제어 장치에 쉽게 영향을 미칠 수 있는 소수의 주 사용자의 선호에 의해 냉난방의 세기 등의 로컬 서비스의 제어가 수동으로 이루어지거나, 또는 이 주 사용자들의 선호만을 학습한 자동 제어 장치가 중앙 집중형으로 로컬 서비스를 제어하였다. 그 결과 사무실, 공연장, 카페 등의 로컬에서 무차별적인 로컬 서비스를 받는 대다수 일반 사용자가 현재의 로컬 서비스 상태를 선호하지 않더라도 이를 함묵적으로 반영하여 조절할 메커니즘이 존재하지 않았다. 이는 로컬 영역의 일반 이용자의 만족도를 현저히 떨어뜨리는 문제를 발생시킬 뿐 아니라, 너무 춥거나 혹은 더운 온도로 설정되어 개인 만족도를 만족시키지 못하면서 에너

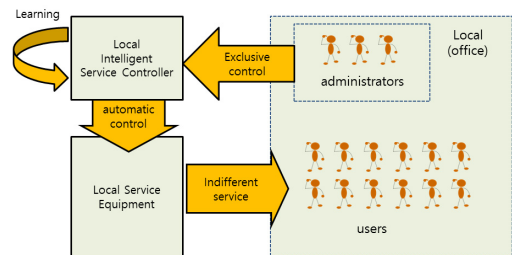


Fig. 1. Centralized local service control model

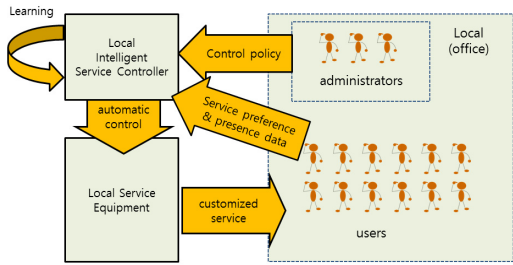


Fig. 2. Individual presence-preference based local service control model

지 낭비로 인한 과도한 비용 문제를 유발할 수 있다. 이러한 중앙집중형 로컬 서비스의 제어 방식에 대한 대안은 로컬 영역 내 일반 사용자의 선호를 중앙 제어 장치에서 수집하여 중앙 제어 장치에서 주 사용자뿐만 아니라 일반 사용자의 선호도를 종합적으로 고려하여 로컬 서비스에 대해 의사결정을 내리는 방법이다 (그림 2 참조).

본 연구에서는 구체적으로 사무실 공간에서의 냉방기 조절 로컬 서비스 시나리오를 가정하여 시스템을 구현한다. 그림 3는 본 연구에서 구현한 개인 프레즌스-선호도 기반 지능형 로컬 서비스 제어 시스템의 전체적인 아키텍처를 보여준다.

IoT 게이트웨이는 지역내의 냉방 서비스를 조절하기 위해 다음의 역할을 수행한다.

- (1) 지역내의 온습도 센서와 에어컨에 부착된 전력 센서를 이용하여 실내 온습도와 에어컨의 전력 사용량을 모니터링한다(분 단위). 그리고 HTTP 프로토콜을 이용하여 외부 기상청의 개방형 API를 이용하여 지역의 실외 온습도 데이터를 수집한다.
- (2) 사용자의 스마트폰에 설치되어 있는 냉방 선호 투표 앱과 통신하여 사용자의 실내 프레즌스 데이터와 사용자의 냉방 선호도를 수집한다.
- (3) 수집된 센서 정보와 사용자의 선호도 정보를 병합하여 주기적(일일)으로 클라우드 상의 지능형 로컬 서비스 서버에 전송한다. 그리고 서버로부터 업데이트된 사용자 냉방 선호 예측 모형을 주기적으로 수신한다.
- (4) 일정 시간 슬롯(15분) 동안 수집된 실내외의 온습도 정보, 시간 슬롯 동안 사용자 프레즌스 정보, 시간 슬롯 내에 선호도 투표를 한 사용자의 선호 투표 정보, 그리고 시간 슬롯 내에 투표하지 않은 사용자의 냉방 선호 예측 정보를 종합하여 다음 4 구간 슬롯 동안의 최적 냉방 조절 시퀀스를 생성

하여 에어컨을 조절한다. 이 때 적외선 발신기를 이용하여 직접 에어컨을 컨트롤 한다.

사용자의 선호를 직접 투표된 데이터뿐 아니라 선호 예측 결과를 동시에 반영하는 이유는 다음과 같다. 처음 학습 기간에는 현재 냉난방에 대한 선호도를 투표 독려를 받지만, 일정 기간이 지나 선호도 학습이 된 후에는 사용자는 투표를 독려 받지 않고 현재의 자동 조절이 불만인 경우에만 의사를 표현하기 때문이다.

사용자 스마트폰에 설치되는 개인 선호 투표 앱은 다음의 기능을 수행한다.

- (1) 사용자는 재실 중인 지역의 IoT 게이트웨이에 자신을 등록한다. 사용자가 여러 사무실에 걸쳐 근무할 경우에는 여러 지역의 게이트웨이에 등록될 수 있다.
- (2) IoT 게이트웨이에 사용자가 등록되면 학습 기간 동안 약 1시간 간격으로 현재 냉방에 대한 선호도 투표 알림을 스마트폰에 표시한다. 선호도 투표는 ASHRAE-55 표준²⁰⁾으로 설정되어 있는 -3부터 3까지의 7점 열쾌적감 선호도 척도를 이용한다.
- (3) 개인 선호 투표 앱은 주기적(10분)으로 자신이 등록된 IoT 게이트웨이에 ping을 수행하며 특정 시간 동안 ping이 없으면 IoT 게이트웨이는 해당 사용자를 부재중으로 파악한다.

클라우드 상의 지능형 로컬 서비스 서버는 다음의 기능을 수행한다.

- (1) 일일 단위로 전달되는 지역의 센서 정보와 사용자 투표 정보를 기반으로 최근 기간 동안의 사용자

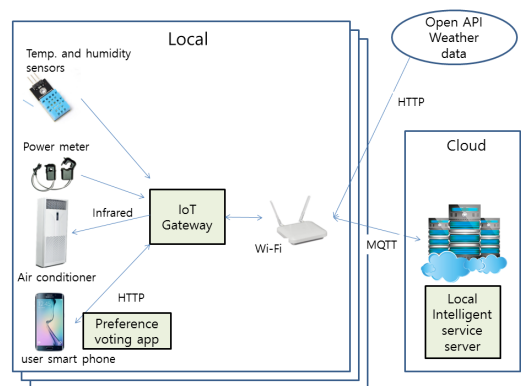


Fig. 3. The architecture of the IPP-LISS

신호도 예측 모형의 정확도를 평가한다.

- (2) 현재의 예측 모형의 예측 정확도가 정해진 기준 이하로 내려가게 되면 최근 데이터를 반영하여 사용자 신호도 예측 모형의 재훈련을 수행한다.
- (3) 기존의 모형과 새롭게 훈련된 모형을 며칠 동안 라이브 상태에서 예측의 정확도를 파악한 후 새롭게 훈련된 모형이 기존의 모형보다 확실히 우위가 있다고 판단되면, IoT 게이트웨이에 새로운 예측 모형을 전송한다.

지능형 로컬 서비스 서버는 한 지역의 IoT뿐 아니라 관할하고 있는 복수의 지역의 IoT 게이트웨이와 데이터와 예측 모형을 주고받는다. 또한 IoT 게이트웨이에 대한 기본적인 모니터링 기능이 있어서 IoT 게이트웨이가 적절히 작동하고 있는지 감시한다.

IV. IPP-LISS 서비스 제어 알고리즘

이 절에서는 IPP-LISS의 서비스 제어 알고리즘을 설명한다. 서비스 제어 알고리즘은 기본적으로 냉난방을 최적으로 조절하는 시퀀스를 발견하는 최적화 알고리즘이다.

그럼 4는 IPP-LISS에서 사용하는 서비스 제어 알고리즘에서 이용하는 기호들의 시간 축 상에서의 관계를 보여준다. 다음은 각 기호에 대한 정의이다.

- $TC(t)$: 시간 구간 $(t, t+1]$ 에서 평균적인 열환경 데이터 벡터를 의미한다. 실내 온습도, 실외 온습도 등의 평균을 포함한다.
- $PMV(t)$: 현재 실내에 있는 재실자의 프레즌스 정보를 바탕으로 시간 구간 $(t, t+1]$ 동안의 실내 사용자의 평균 열쾌적감 예측 값이다.
- $d(t)$: 시간 구간 $(t, t+1]$ 시점동안 냉방기의 조절에 대한 의사결정이다. 현재 모형에서는 설정 온도만 의사결정을 수행한다. 그리고 이 기간 동안 냉방기의 조절은 유지된다고 가정 한다.
- $p(t)$: 시간 구간 $(t, t+1]$ 동안 냉방기에 의해 사용되는 에너지 사용량이다.
- $\widehat{A}(t)$: 함수 $A(t)$ 값에 대한 추정치를 나타낸다.

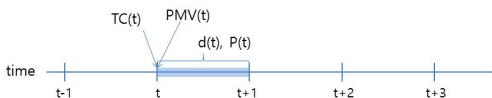


Fig. 4. Notation

그러면 전력 사용량 예측 모형은 결국 다음과 같이 표현할 수 있다.

$$\widehat{P}(t) = h_p(d(t), TC(t-1)) \quad (1)$$

$$\widehat{TC}(t) = f_{TC}(d(t), TC(t-1)) \quad (2)$$

열쾌적감 예측 모형은 다음과 같이 표현할 수 있다.

$$\widehat{PMV}(t) = g_{PMV}(TC(t-1), TC(t-2), \dots, TC(t-l)) \quad (3)$$

본 연구에서는 데이터 마이닝 기법을 이용하여 g_{PMV} 를 함수를 추정하고 h_p 와 f_{TC} 함수는 이론적 추론을 통해 유도한다. h_p 는 현재 온도와 설정 온도의 차이에 의해 선형적으로 증가하는 함수를 가정하였고 냉방기의 최대 냉방부하까지 증가하고, 그 이후에는 최대 냉방부하를 유지하는 함수를 가정하였다. f_{TC} 는 현재의 온도와 냉방기에서 나오는 공기의 양과 온도를 이용하여 열평형 방정식을 이용하여 도출하였다. 문제의 단순화를 위하여 습도, 복사열 등의 효과는 일정하다고 가정하였다.

서비스 제어 알고리즘은 최적화 알고리즘으로서 목적 함수는 m 시간 슬롯 기간 동안의 사용자의 열쾌적감을 최대화하고 전력 사용량을 최소화하는 것을 목적으로 하도록 구성된다. $C(t)$ 를 $(t, t+1]$ 기간 동안에 전력 사용과 실내 열환경에 따른 불쾌적감에 의해 발생하는 비용이라고 정의하자. 본 연구에서는 $C(t)$ 를 다음과 같이 정의한다.

$$C(t) = c_p h_p(t) + \lambda |g_{PMV}(t)|^\beta, \lambda > 0, \beta > 0. \quad (4)$$

단, λ 는 불쾌적감을 비용화하는 계수이고, β 는 불쾌적감의 차원 척도이다. 그리고 c_p 는 단위 전력 사용량의 비용 계수이다. (4)에서 우리는 다중 기준 최적화 함수를 단일 기준 최적화 함수로 변형하는 가장 간단한 방식은 선형 조합 방식을 이용하였다. [21] 그러면 시점 t 부터 미래 m 시간 슬롯 동안 발생하는 전체 비용 함수 $C(t, m)$ 은 다음과 같이 정의할 수 있다.

$$C(t, m) = \sum_{j=0}^{m-1} \alpha^j C(t+j), \quad 0 < \alpha \leq 1. \quad (5)$$

단, α 는 미래의 불확실성에 따른 할인율이다. 시스

템 관리자는 α, β, λ 등의 튜닝 매개변수를 바꾸어가며 지역 특성을 고려한 최적화를 시도할 수 있다. 본 연구에서는 한 시간 슬롯의 기간이 15분으로 하고, $m = 4$ 로 하여 1시간 동안의 비용 발생을 최적화하도록 냉난방을 조절한다.

그림 5는 사용자의 열쾌적감 선호도 예측 모형을 생성하기 위한 IPP-LISS 서버의 구성 요소를 보여준다.

데이터 수집기(Data Collector)는 IoT 게이트웨이로부터 센서 정보와 사용자의 열쾌적감 투표 정보를 수집한다.

데이터 전처리기(Data Preprocessor)는 수집된 데이터를 이용해 다음 작업을 수행한다.

- (1) 수집된 센서 데이터와 사용자 열쾌적감 선호 투표 결과를 시간 슬롯을 기준으로 병합한다.
- (2) 데이터의 이상치를 검출하여 가능하면 데이터의 수정을 시도한다.
- (3) 기존의 예측 변수로부터 유도되는 새로운 예측 변수를 생성한다.

데이터 전처리기에서 사용자 열쾌적감 예측 모형을 위한 모형 집합(model set)이 전달되면 모형 생성기(Model Generator)는 다음 작업을 수행한다.

- (1) IoT 게이트웨이로부터 수집된 데이터를 훈련/검증/테스트 집합으로 분리한다.
- (2) 교차 검증의 방법으로 데이터 마이닝 기법의 매개 변수를 다양하게 변경하면서 다양한 예측 모형을 생성한 후 성능을 점검한다.
- (3) 검증 집합에서 가장 최고의 성능을 보인 매개 변수를 선택하여 모형을 최적화한다.

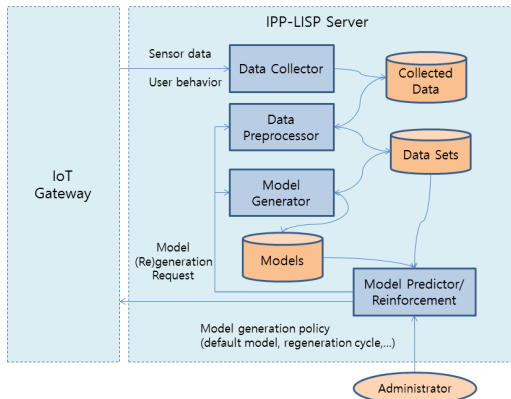


Fig. 5. Architecture of the IPP-LISS server

- (4) 최적화된 모형을 테스트 집합에 적용하여 최종 예측 성능을 추정한다.

열쾌적감 예측 모형을 개발하기 위해 다음의 데이터 마이닝 기법을 사용한다.

- 의사결정나무 분류 모형과 회귀 모형
- 랜덤 포레스트 분류 모형과 회귀 모형
- 인공신경망 분류 모형과 회귀 모형

각 데이터 마이닝 기법에 대한 설명은 Kuhn and Johnson^[22]을 참조하라.

분류 모형은 현재 열환경에 대한 사용자의 선호도를 cold, neutral, hot 3 가지 범주로 나누어 변환한 후 예측하는 모형이고, 회귀 모형은 사용자의 평균적 선호도 수치를 예측하는 모형이다. 본 연구에서는 이 두 가지 방식으로 모두 예측 모형을 개발하였다. 분류 모형의 성능은 다음 3가지 지표를 이용하여 평가하였다.

- (1) 오분류율: 실제 클래스와 다른 클래스로 예측하는 비율.
- (2) 추천 정확도: 분류 모형이 hot 또는 cold를 추천할 때 정확히 맞는 비율.

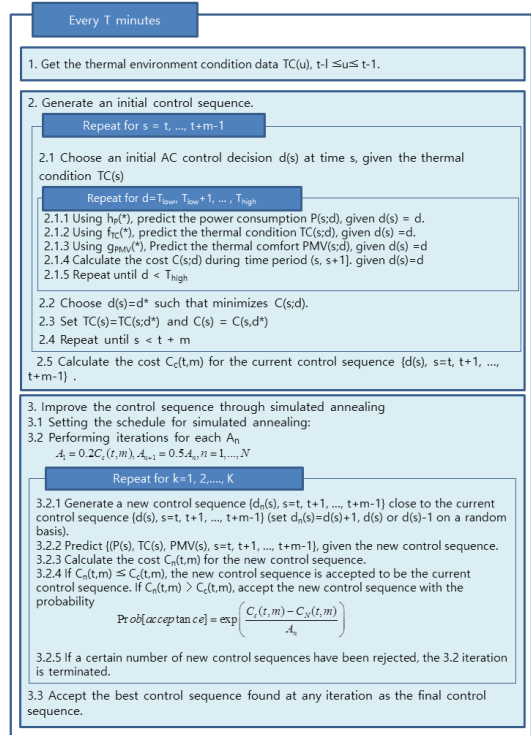


Fig. 6. Optimal service control algorithm

(3) 추천 재현도: 실제 값이 hot 또는 cold일 때 추천 모형이 미리 예측해 준 비율.

반면 회귀 모형은 모형의 예측 값을 범주형 변수로 변환한 후 분류 모형과 마찬가지로 오분류율, 추천 정확도, 추천 재현도를 확인하였고, 회귀 모형의 수치 예측 값과 실제 값의 차이를 평가하기 위해 RMSE(root mean square error)를 같이 계산하였다.

데이터 마이닝 기법을 이용하여 관련 함수에 대한 추정치가 완료되면 최적화 기법을 이용하여 최적 냉방 조절 시퀀스를 생성한다. 그림 6는 최적화 모형을 이용하여 냉방기에 대한 최적 제어 시퀀스를 생성하는 알고리즘의 개요를 보여준다. 알고리즘은 크게 3 부분으로 나뉘어 매 15분마다 실시된다.

- (1) 알고리즘의 첫 번째 부분은 열환경 정보를 받아오는 부분이다.
- (2) 두 번째 부분에서 향후 $m = 4$ 시간 슬롯에서 비용이 최소화되는 냉방 조절 시퀀스를 그리디 알고리즘으로 생성한다. 이렇게 생성된 최초의 자동 조절 시퀀스는 각 슬롯에서의 최적화만을 고려한 것이므로 전역 최적해가 아닐 수 있다.
- (3) 따라서 우리는 세 번째 단계에서 simulated annealing 메타휴리스틱 방법^[23]을 이용하여 더 나은 냉방 조절 시퀀스를 찾아 나간다.

V. IPP-LISS 프로토타입 구현 결과

본 장에서는 앞서 설명한 IPP-LISS의 아키텍처와 알고리즘에 따라 프로토타입 시스템을 구현한 결과를 제시한다.

그림 7은 실제 구현된 IoT 게이트웨이의 내부 구조

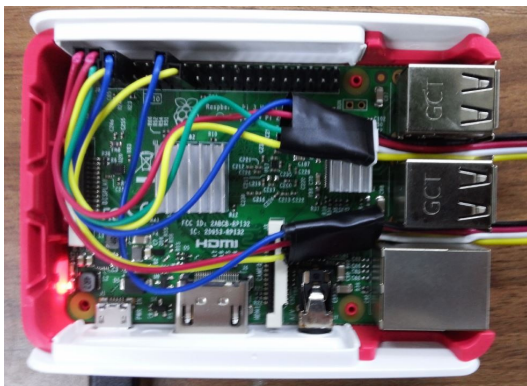


Fig. 7. IoT gateway

를 보여준다. IoT 게이트웨이는 라즈베리 파이로 구현되었고 구체적인 명세는 다음과 같다.

- 모델명: Raspberry Pi 3 Model B.
- CPU: ARMv8(BCM2837) 64bit Quad Core 1.2GHz.
- RAM: 1GB.
- 연결방식: 802.11n Wireless LAN, Ethernet, Bluetooth 4.1, 4 USB ports, 40 GPIO pins, Full HDMI port.

또한 IoT 게이트웨이에는 다음 센서가 부착되었다.

- 온습도 센서: Grove - Temperature & Humidity Sensor (TH02)
- IR 송신기: Grove - Infrared Emitter.

표 1은 게이트웨이-월별 데이터가 수집된 현황이다. 총 15 개의 게이트웨이에서 데이터가 수집되었고, 표에서 보듯이 게이트웨이별로 편차들이 있지만 50% 정도의 데이터가 9월에 수집되었고 8월과 10월에 약 25% 정도의 데이터가 수집되었다.

표 2는 각 게이트웨이에서 수집된 사용자의 열쾌적감 투표의 현황을 보여준다.

표 3은 의사결정나무, 랜덤 포레스트, 인공신경망

Table 1. Sensor data collecting frequency for each gateway-month combinations

gateway	Aug.	Sep.	Oct.
11c85b	109523	171395	38610
1f1bf9	114211	172365	105611
276bac	36574	117583	0
2c8488	50696	0	0
311620	7676	54881	0
3cd48b	7553	82029	33758
62f77a	76165	87874	113565
75d9de	12	0	0
7fd007	77369	171115	29424
9a7cd1	34	0	0
a54379	77561	79779	116313
bacef8	35685	171296	105439
c4fc0b	6100	263	0
d2e5fc	70	0	0
f6c9fe	110331	172680	117270
SUM	709560	1281260	659990
Prop.(%)	27	48	25

Table 2. Summary statistics of thermal comfort votes for each gateway

	number of voters	number of votes	min of votes per voter	max of votes per voter	mean of votes per voter
1f1bf9	6	1413	51	391	236
a54379	5	351	4	166	70
62f77a	5	294	2	194	59
11c85b	2	123	1	122	62
c4fc0b	3	103	16	49	34
bacef8	2	92	2	90	46
f6c9fe	2	77	25	52	38
276bac	1	67	67	67	67
2c8488	4	26	3	10	6
7fd007	3	23	4	14	8
3cd48b	1	14	14	14	14
311620	2	9	2	7	4

Table 3. Error rate, precision and recall for each data mining techniques

	error rate	precision	recall
tree-clas	0.346	0.647	0.566
tree.reg	0.424	0.618	0.360
RF.cls	0.333	0.679	0.588
RF.reg	0.338	0.695	0.566
NN.clas	0.355	0.654	0.566
NN.reg	0.368	0.641	0.564

분류 및 회귀 모형의 오분류율, 추천 정확도, 추천 재현도를 보여준다. 사실 회귀 나무를 제외하고는 매우 비슷한 성능을 보여줌을 알 수 있다. 오분류율 측면에서는 랜덤 포레스트의 분류 모형이, 추천 정확도 측면에서는 랜덤 포레스트의 회귀 모형이, 재현도는 랜덤 포레스트의 분류 모형이 근소한 차이로 더 좋았다.

냉방기의 최적 조절 시퀀스를 구하기 위해서는 목적 함수(비용 함수)의 형태를 결정해야 한다. 비용 함수를 결정하기 위해서는 전력 사용량에 대한 단위 비용과 사용자의 열쾌적 불만족에 대한 벌금 계수를 정해야 한다. 실제 이 값의 설정은 실제 문제에서 관리자가 시뮬레이션을 통하여 결정해야 한다. 여기서는 전력 사용량의 단위 비용 $c_p=1$ 로 하고, 불만족 정도에 대한 벌금 항목 계수 $\lambda=200$ 로, 불만족 정도에 대한 차원 $\beta=2.5$ 로 정하여 최적 냉방 시퀀스를 찾는 시뮬레이션을 수행하였다.

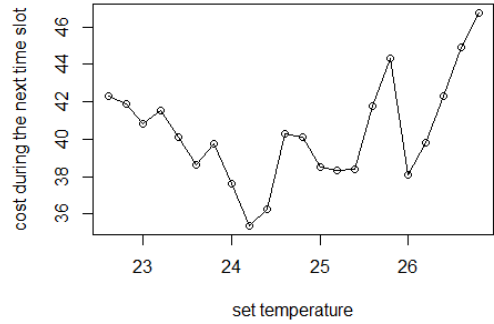


Fig. 8. Simulation result of the cost function at various set temperatures

그림 8는 실내 온도가 26°C, 실내 상대 습도가 60%, 실외 온도가 29°C, 실외 상대 습도가 68%일 때, 에어컨 설정 온도를 바꾸어가며 비용 함수의 시뮬레이션 결과를 보여준다. 설정온도가 내려가면 전력 사용량의 비용 때문에 전체 비용이 상승하고, 설정 온도가 올라가면 불쾌적감에 따른 벌금 항목 때문에 비용이 올라간다. 이 예에서는 24°C, 근방에서 한 시간 슬롯의 비용이 최소가 됨을 볼 수 있다.

표 4는 그리디 알고리즘을 이용하여 냉방 조절에 대한 초기 시퀀스를 얻은 다음, simulated annealing 방법을 사용하여 냉방 조절 시퀀스가 개선되는 모습을 보여준다. 전체적으로 급격하지 않은 방식으로 온도를 조절하는 것을 더 좋은 해로 찾는 것으로 보인다.

Table 4. An example of improving a service control sequence with the simulated annealing method

	decision slot1	decision slot2	decision slot3	decision slot4	cost
1	24	24.0	24.0	24.0	150.45
2	24	22.5	22.5	25.5	148.54
3	24	22.5	22.5	25.5	148.54
4	24	22.5	22.5	25.5	148.54
5	24	22.5	22.5	25.5	148.54
6	24	22.5	22.5	25.5	148.54
7	24	22.5	22.5	25.5	148.54
8	24	22.5	22.5	25.5	148.54
9	24	22.5	22.5	25.5	148.54
10	24	22.5	22.5	25.5	148.54
11	24	22.5	22.5	25.5	148.54
12	24	22.5	22.5	25.5	148.54
13	24	22.5	22.5	25.5	148.54
14	24	22.5	22.5	25.5	148.54

VI. IPP-LISS에 MEC 기술 적용 방안

IPP-LISS 시스템이 지향하는 지능형 로컬 서비스를 하기 위해서는 IoT 게이트웨이는 지능을 가져야 하고 이를 위해서는 계산 집약적인 모형 생성 작업을 수행해야 한다. 그러나 IPP-LISS 시스템의 IoT 게이트웨이는 일반적인 컴퓨터에 비해 낮은 컴퓨팅 용량과 작은 저장 공간을 가지고 있다. 이러한 문제를 해결하는 좋은 방법 중 하나가 MEC 서버에 모형 생성 작업 등을 오프로딩하는 것이다.

표 5는 IPP-LISS의 프로토타입 구현을 통해서 측정된 각 작업의 컴퓨팅 및 저장 요구사항이다.

(1) 출력 저장 요구사항(output storage requirement)은 특정 IoT 게이트웨이에서 6주 동안 수집된 데이터에 대하여 데이터 수집, 데이터 사전처리, 모형 생성, 최적 냉방 조절 시퀀스 결정의 작업을 수행한 후, 그 작업의 출력 객체의 저장 용량을 측정 한 결과이다. 각 단계의 출력 객체는 바로 다음 단계의 입력 객체가 된다.

(2) 처리 시간(processing time)은 각 작업의 처리에 든 시간이다. 같은 스케일로 측정하기 위해 IoT 게이트웨이의 작업도 클라우드 서버 상에서 수행하여 측정을 수행하였다. 참고로 클라우드 상의 서버의 사양은 다음과 같았다.

CPU: Intel Xeon E5-1650 v3 3.50 GHz

RAM: 32GB

OS: Windows7 Professional K service pack 1

(3) 작업 빈도(frequency)는 각 작업이 얼마나 빈번하게 일어나는지를 나타내고, 아울러 각 작업의 latency 요구사항도 표시하였다.

IoT 기기와 MEC 서버가 LTE망에서의 메모리 객체를 주고받는 작업이 매우 큰 병목 지점이 될 것으로 예상된다. 왜냐하면 LTE망은 높은 throughput을 제공하나 낮은 latency 요구사항의 응용에 최적화되어 있는데, 일반적으로 IoT 기기의 응용은 낮은 throughput에 높은 latency 요구사항을 가지고 있기 때문이다.

이러한 고려 사항과 표 5의 결과를 기준으로 IPP-LISS의 작업을 다음과 같이 분할하여 수행하는 것이 적절할 것으로 판단된다.

(1) IoT 게이트웨이에서 직접 수행할 작업

데이터 수집 작업: 낮은 빈도와 높은 latency 요구사항이 있어 IoT 게이트웨이에서 처리하는 것이

Table 5. Storage and computing requirements of each job for IPP-LISS

Job	output storage requirement	processing time	frequency	latency requirement	
Data collecting	1,455KB	-	every minute	< 1sec	
Data preprocessing	38KB	5s	every day	< 2 hours	
Model generation	tree-clas	572KB			0.7s
	tree.reg	626KB			0.3s
	RF.cls	20,790KB			800.71s
	RF.reg	38,429KB			239.45s
	NN.clas	12,133KB			2784.08s
NN.reg	6,865KB	2716.09s			
Optimal decision	1KB	21s	every 15 min.	< 3 min	

적절하다.

데이터 전처리 작업: 원본 데이터에서 모형 집합 데이터를 생성하는 작업이다. 처리 시간이 어느 정도 소요되지만 낮은 latency 요구사항을 갖는다. 또한 데이터 수집 작업 후 이를 LTE 망으로 전송하는 것에 비해 전처리한 후의 데이터를 LTE 망을 통해 전송하는 것이 데이터 크기가 2.6%로 크게 줄어들므로 망 전체의 효율성을 높일 수 있다.

(2) MEC 서버에서 오프로딩으로 처리할 작업

모형 생성 작업은 긴 처리 시간, 큰 저장 공간, 낮은 latency 요구사항을 가지므로 MEC 서버에 오프로딩하여 수행하는 것이 적절하다.

(3) 추후 연구를 통해 판단이 필요한 작업

최적 냉방 조절 시퀀스를 정하는 작업은 15분 간격 정도로 발생하니 어느 정도 자주 발생하는 작업이므로 IoT 게이트웨이에서 처리되는 것을 일차적으로 고려할 수 있다. 그러나 처리 시간이 꽤 길고, 이 작업에 필요한 입력 객체인 모형이 랜덤 포레스트나 인공신경망을 사용하는 경우 10~20MB 정도의 크기를 가지고 있어 추후 시뮬레이션 연구 등을 통해서 어느 방식이 적합한지 판단되어야 할 것이다.

VII. 결 론

이 논문에서는 개인 프레즌스-선호도 기반 지능형 로컬 서비스 제어 시스템(IPP-LISS)을 위한 서비스 시나리오와 아키텍처를 제시하고, 서비스 제어 알고리즘을 개발한 후, 프로토타입 시스템을 개발하였다. 아

올러 IPP-LISP 시스템을 모바일 엣지 컴퓨팅 환경에 적용하는 방안에 대하여 논의하였다.

이러한 연구를 통해 다음과 같은 시사점을 얻었다.

첫째, 지능형 서비스를 위한 예측 모형을 생성하는 데이터 마이닝 기법들은 정확도 측면에서는 랜덤 포레스트와 인공신경망은 매우 비슷한 성능을 보였다. 그러나 랜덤 포레스트에 비해 인공 신경망은 훈련 시간이 긴 단점을 보였고, 랜덤 포레스트는 인공 신경망에 비해 생성된 모형의 저장 공간이 큰 단점을 보였다.

둘째, 냉방 조절 시퀀스 생성 알고리즘은 매우 빠르게 지역 최적해로 수렴하는 경향을 보였다. 지속적인 반복에도 크게 지역 최적해가 변하지 않는 모습을 보인다. 물론 본 연구가 다양한 세팅 하에서 최적해를 탐색하는 부분이 생략되어 한계를 가진 결론이지만 5장에 보인 사례만으로 매우 빠르게 지역 최적해로 수렴함을 알 수 있다.

셋째, IPP-LISP 시스템의 각 작업의 빈도, 저장 공간 요구량, 처리 시간, latency 요구 등을 분석해본 결과 IPP-LISP은 MEC 환경의 매우 유력한 오프로딩 서비스 시나리오라는 것을 확인하였다.

앞서 밝힌 연구의 시사점에 근거하여 다음과 같은 향후 연구 방향을 제시한다.

첫째, 추후 연구에서는 비선형 회귀모형 등 좀 더 다양한 데이터 마이닝 기법을 적용하여 모형의 정확도, 모형 생성의 처리시간, 모형의 저장 공간 측면에서 최적의 되는 분석 기법을 찾는 연구가 필요하다.

둘째, 지역 최적해로 빠르게 수렴하는 문제를 해결하기 위해 자동 조절 냉방 시퀀스를 단일 온도 세팅, 천천히 온도 세팅 변화, 빠른 온도 세팅 변화 등의 다양한 초기 조절 시퀀스를 생성한 후 최적해를 탐색해 보는 것이 필요하다. 이를 위해서는 하나의 경로로 최적해를 찾는 simulated annealing 방법보다는 유전자 알고리즘 등의 모집단 기반 메타휴리스틱 방법이 유력해 보인다.

셋째, 서비스 최적 시퀀스 생성 알고리즘을 MEC 환경에서 오프로딩하는 것의 적절성을 파악하기 위하여 실제 또는 시뮬레이션 환경 하에서 각 작업의 오프로딩 여부에 따른 성능 지표를 파악하는 추후 연구가 필요하다.

References

[1] M. T. Beck, M. Werner, S. Feld, and S. Schimper, "Mobile edge computing: A taxonomy," in *Proc. 6th Int. Conf. Advances*

in Future Internet, 2014.

[2] ETSI, *Mobile-edge computing –Introductory technical white paper*, ETSI White Paper, 2014.

[3] A. Ahmed and E. Ahmed, "A survey on mobile edge computing," in *Proc. 10th IEEE Int. Conf. Intell. Syst. and Contr.*, Coimbatore, India, 2016.

[4] C.-W. Tsai, C.-F. Lai, M.-C. Chiang, and L. T. Yang, "Data mining for internet of things: A survey," *IEEE Commun. Surveys & Tuts.*, vol. 16, no. 1, pp. 77-97, 2014.

[5] S. D. T. Kelly, N. K. Suryadevara, and S. C. Mukhopadhyay, "Towards the implementation of iot for environmental condition monitoring in homes," *Sensors J. IEEE*, vol. 13, no. 10, pp. 3846-3853, 2013.

[6] Y. Zeng, Z. Zhang, and A. Kusiak, "Predictive modeling and optimization of a multi-zone hvac system with data mining and firefly algorithms," *Energy*, vol. 86, pp. 393-402, 2015.

[7] H.-C. Jo, S. Kim, and S.-K. Joo, "Smart heating and air conditioning scheduling method incorporating customer convenience for home energy management system," *IEEE Trans. Consumer Electron.*, vol. 59, no. 2, pp. 316-322, 2013.

[8] M. M. Tehrani, Y. Beauregard, M. Rioux, J. P. Kenne, and R. Ouellet, *A predictive preference model for maintenance of a heating ventilating and air conditioning system*, IFAC-PapersOnLine, vol. 48, no. 3, pp. 130-135, 2015.

[9] G. Zucker, J. Malinao, U. Habib, T. Leber, A. Preisler, and F. Judex, "Improving energy efficiency of buildings using data mining technologies," *IEEE ISIE*, pp. 2664-2669, 2014.

[10] M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, "The case for vm-based cloudlets in mobile computing," *IEEE Pervasive Comput.*, vol. 8, no. 4, pp. 14-23, 2009.

[11] K. Habak, M. Ammar, K. A. Harras, and E. Zegura, "Femto clouds: Leveraging mobile

devices to provide cloud service at the edge,” in *2015 IEEE 8th Int. Conf. Cloud Comput.*, pp. 9-16, 2015.

[12] S. Abdelwahab, B. Hamdaoui, M. Guizani, and T. Znati, “Replisom: Disciplined tiny memory replication for massive IoT devices in LTE edge cloud,” *IEEE Internet of Things J.*, vol. 3, no. 3, pp. 327-338, 2016.

[13] M. T. Beck, S. Feld, A. Fichtner, C. Linnhoff-Popien, and T. Schimper, “Me-volte: Network functions for energy-efficient video transcoding at the mobile edge,” in *18th Int. Conf. Intell. Next Generation Netw.*, pp. 38-44, 2015.

[14] X. Chen, L. Jiao, W. Li, and X. Fu, “Efficient multi-user computation offloading for mobile-edge cloud computing,” *IEEE/ACM Trans. Networking*, vol. 24, no. 5, pp. 2795-2808, 2015.

[15] T. Guettari, “Thermal signal analysis in smart home environment for detecting a human presence,” *Int. Conf. Advanced Technol. for Sign. and Image Process.*, pp. 334-339, 2014.

[16] S. Lee, S. Y. Jeong, S. J. Kang, and W. J. Lee, “Design and implementation of IoT chatting service based on indoor location,” *J. KICS*, vol. 39, no. 10, pp. 920-929, 2014.

[17] S. D’Oca and T. Hong, “A data-mining approach to discover patterns of window opening and closing behavior in offices,” *Build. and Environ.*, vol. 82, pp. 726-739, 2014.

[18] H. K. Jung, S. Jung, D. H. Lee, S. Q. Lee, and J.-H. Kim, “Wireless caching algorithm based on user’s context in smallcell environments,” *J. KICS*, vol. 41, no. 7, pp. 789-798, 2016.

[19] S. Y. Jeon, J. H. Ahn, and T.-J. Lee, “Broadcast data delivery in iot networks with packet loss and energy constraint,” *J. KICS*, vol. 41, no. 2, pp. 269-276, 2016.

[20] ASHRAE, “Standard 55-thermal environmental conditions for human occupancy,” *ASHRAE Standard*, vol. ASHRAE-55, 2013.

[21] E. K. Chong and S. H. Zak, *An introduction*

to optimization, vol. 76. John Wiley & Sons, 2013.

[22] M. Kuhn and K. Johnson, *Applied predictive modeling*, Springer, 2013.

[23] E.-G. Talbi, *Metaheuristics: From design to implementation*, John Wiley & Sons, 2009.

김 길 환 (Kilhwon Kim)



1994년 2월 : KAIST 경영과학
과 졸업

1996년 2월 : KAIST 산업경영
학과 석사

2009년 2월 : KAIST 산업및시
스템공학과 박사

1998년 11월~2003년 3월 : LG
CNS 선임컨설턴트

2003년 4월~2005년 3월 : 기업정보화지원센터 선임
연구원

2009년 1월~2012년 8월 : ETRI 선임연구원

2012년 8월~현재 : 상명대학교 경영공학과 조교수

<관심분야> 시스템 성능 분석, 대기행렬 이론, 서비
스 플랫폼, 데이터 마이닝

장 진 산 (Jin-San Jang)



2015년 8월 : 상명대학교 경영
공학과 졸업

2015년 9월~현재 : 상명대학교
경영공학과 석사과정

<관심분야> 데이터 마이닝, 빅
데이터 분석

금 창 섭 (Changsup Keum)



1992년 2월 : 서울시립대학교 전산통계학과 학사

2005년 8월 : 카네기멜론 대학 소프트웨어 공학 석사

2013년 8월 : 한국과학기술원 공학박사

1994년 2월~현재 : 한국전자통신연구원 책임연구원

<관심분야> 5G 모바일 클라우드, 엣지 서비스 플랫폼, 초연결 분산 신뢰 서비스, 소프트웨어 아키텍처

정 기 숙 (Ki-Sook Chung)



1995년 2월 : 포항공과대학교 컴퓨터공학과 졸업

1997년 2월 : KAIST 전산학과 석사

1997년 1월~2001년 9월 : 데이콤 종합연구소 연구원

2001년 10월~현재 : ETRI 책임연구원

<관심분야> 초연결 네트워크 서비스, 모바일 엣지 컴퓨팅, 소프트웨어 아키텍처, 서비스플랫폼