

스마트 팩토리 환경에서 클라우드와 학습된 요소 공유 방법 기반의 효율적 엣지 컴퓨팅 설계

황지온*

Design of Efficient Edge Computing based on Learning Factors Sharing with Cloud in a Smart Factory Domain

Zi-on Hwang*

Department of Smart Systems Software Engineering, Hyupsung University, Hwaseong 18330, Korea

요 약

최근 사물인터넷은 인공지능의 발전, 연결된 기기의 증가와 클라우드 시스템의 높은 성능으로 인해 급격하게 발전하고 있다. 많은 기기와 센서로부터 생산되는 엄청난 양의 데이터들은 지능적 진단, 추천 서비스 뿐 아니라 스마트 관제 서비스와 같이 서비스 영역의 확대를 이끌고 있다. 엣지 컴퓨팅(Edge Computing)에 대한 연구는 높은 성능을 지닌 하드웨어를 바탕으로 작은 또 하나의 서버로써의 역할에 국한되어 연구되고 있다. 그러나 데이터를 분석하고 의미성에 따른 서비스를 구현하기 위해서는 범용적 서버로써의 역할보다는 도메인에 특화된 기능과 요구사항을 지녀야 한다. 스마트 팩토리에서의 엣지는 제한적 필터링, 사전 포매팅을 포함하는 전처리와 그룹 컨텍스트 융합, 지역적 룰의 관리 등을 필요로 한다. 따라서 본 논문에서는 공장 특성에 맞는 효율성과 강건함 측면을 강조하는 요구사항들을 도출하고, 클라우드와 학습된 요소 공유 방법을 기반으로 하는 엣지 컴퓨팅의 구조를 제안하고자 한다. 이 엣지는 네트워크 자원 소모를 감소시키고 룰과 학습화된 모델의 변경을 쉽게 할 수 있도록 한다.

ABSTRACT

In recent years, an IoT is dramatically developing according to the enhancement of AI, the increase of connected devices, and the high-performance cloud systems. Huge data produced by many devices and sensors is expanding the scope of services, such as an intelligent diagnostics, a recommendation service, as well as a smart monitoring service. The studies of edge computing are limited as a role of small server system with high quality HW resources. However, there are specialized requirements in a smart factory domain needed edge computing. The edges are needed to pre-process containing tiny filtering, pre-formatting, as well as merging of group contexts and manage the regional rules. So, in this paper, we extract the features and requirements in a scope of efficiency and robustness. Our edge offers to decrease a network resource consumption and update rules and learning models. Moreover, we propose architecture of edge computing based on learning factors sharing with a cloud system in a smart factory.

키워드 : 엣지 컴퓨팅, 학습된 요소 공유, 인스턴트 결정, 스마트 팩토리

Key word : Edge Computing, Learning Factor Sharing, Instant decision-making, Smart Factory

Received 08 September 2017, Revised 08 September 2017, Accepted 21 September 2017

* Corresponding Author Zi-on Hwang(E-mail:zhwang@uhs.ac.kr, Tel:+82-31-299-0908)

Department of Smart Systems Software Engineering, Hyupsung University, Hwaseong 18330, Korea

Open Access <https://doi.org/10.6109/jkiice.2017.21.11.2167>

print ISSN: 2234-4772 online ISSN: 2288-4165

©This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.
Copyright © The Korea Institute of Information and Communication Engineering.

I. 서 론

20세기 초반 유비쿼터스 컴퓨팅에 대한 연구는 집 밖에서 맥 내 기기에 대한 모니터링과 제어, 집 안에서 음성을 통한 기기 관리가 가능하도록 서비스를 제안하였다. 이러한 서비스의 기반은 사용자와 기기에 대한 모니터링, 트래킹, 상황 관리이지만 이를 위한 센서 네트워크를 설치하는 비용적인 문제와 단순한 기기(Thin Client)로 인한 서비스의 제약 문제로 인해 확산되지 못하였다. 그러나 사물인터넷의 확산으로 기기들은 내부에 센서를 포함하거나 미약하지만 일부의 지능화 기능을 내포하게 되었다. 그리고 이렇게 기기와 센서가 증가하며 발생하는 수많은 데이터가 사용자와 기기, 환경에 대한 정보를 수립 가능하게 만들어주고 있다. 데이터의 증가는 인공지능과 빅 데이터 기술을 사물인터넷에 접목하도록 하였다.

전 세계적으로 사물인터넷 플랫폼을 선점하기 위한 노력과 연구가 진행되고 있다. 이미 수백 개에 이르는 플랫폼이 등장하고 이를 통해 센서와 기기를 연결하고 정보를 통한 서비스를 강화하려 하고 있다. 이러한 기술적 성장은 아마존의 AWS나 마이크로소프트의 Azure와 같이 클라우드 영역과 그 내부에서 필요한 기술 세트를 제공하는 인프라 사업적 부분과 IBM Watson IoT나 GE의 Predix와 같이 독립적 사물인터넷 플랫폼을 구축하고 여기에 서비스와 영역을 넓혀나가는 생태계 사업적 부분으로 구분될 수 있다. 다른 측면에서 사물인터넷 기술은 데이터를 수집하고 이를 클러스터링, 개인화를 통해 그룹 또는 개인에게 적합한 서비스 제공을 목적으로 하는 B2C에서의 영역과 수많은 데이터로부터 의미성을 도출하고 도메인 특성에 맞게 가공하여 새로운 서비스를 발굴하는 것을 목적으로 하는 B2B에서의 영역으로 구분할 수도 있다.

그림 1은 최근 대두되고 있는 도메인적 특성을 지닌 서비스들에 대한 구분과 해당 영역에서의 아키텍처를 보여주고 있다. 유비쿼터스 시대에서부터 항상 중요한 서비스 도메인이었던 스마트 홈부터 최근 가장 이슈가 되고 있으며, 산업 전반에 걸친 커다란 변화인 4차 산업혁명의 중심에 있는 스마트 팩토리, 그리고 무인 자동차에 대한 연구에 Google, Tesla와 자동차 OEM들, 서비스 제공업체들이 뛰어들며 급격한 발전을 이루고 있는 지능형 차량들이 주요 도메인으로 부각되고 있다.

그리고 이를 모두 아우르고 연결하여 유통과 흐름을 통제, 관리, 운영하기 위한 스마트 시티 또한 커다란 연구 분야가 되고 있다.

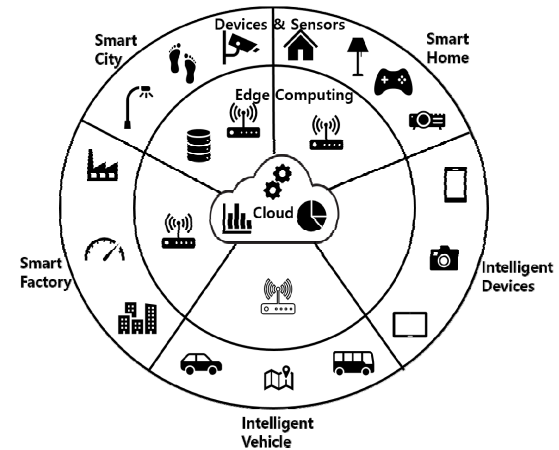


Fig. 1 IoT Services and Architectures

지능형 기기 영역에서, 앞서 언급한 B2C에서의 개인화된 서비스는 기기나 센서가 직접적으로 클라우드로 접속하여 데이터를 제공하고 서비스를 받으면 되도록 되어 있다. 삼성이나 LG와 같은 모바일 기기 제조사나 Google, Apple과 같은 OS와 클라우드를 보유하고 있는 서비스 사업자 모두 기기와 클라우드를 연결하여 추천 서비스를 제공하도록 하고 있다. 그리고 지능형 차량 영역에서는 차량 업체의 경우 중간에 내부 서버를 두고 외부 클라우드와 연계하는 형태와 직접적으로 외부 클라우드로 연계하는 형태를 데이터의 특성에 따라 오케스트레이션(Orchestration)하는 구조를 띄기도 한다. 다만 이 경우 내부 서버가 일부 엣지 컴퓨팅의 역할을 수행하기도 한다.

반면 스마트 시티나 스마트 홈의 경우 서비스 제공 업체나 기기 제조사가 제공하는 게이트웨이를 통해 기기가 연결되는 형태를 지니며 이 게이트웨이가 엣지 컴퓨팅의 역할을 수행한다. 스마트 팩토리의 경우 공장 내의 넓은 공간적 특성, 공간별로 다른 역할을 담당해야 하는 지역적 특성, 수많은 기기와 센서에서 생성되는 데이터의 양적 특성이 결합되어 가장 엣지 컴퓨팅을 필요로 하는 영역이 된다. 또한 유사한 역할을 하는 센서나 기기들이 지역적으로 묶여 있기 때문에 패터이나

특성 분류에 따른 서비스 제공이 가능하도록 요구되기도 한다.

본 논문은 이에 따라 엣지 컴퓨팅에서의 중요한 도메인이 되는 팩토리 분야에서 요구되는 요구사항을 검토하고 이에 대한 기술적 분류를 수행하며, 효율적이고 최적화된 서비스를 제공하기 위한 방안을 아키텍처를 기반으로 제시한다. 그리고 지역적 특성에 따른 패턴화를 고려하여 사용자가 효율적 기기 관리를 하기 위한 룰 적용 및 생성 방안과 데이터에 대한 지능적 필터링, 학습에 기반을 둔 처리를 제공할 수 있는 Instant Decision-Making 제공 방법을 제시한다.

II. 본 론

2.1. 관련 연구

엣지 컴퓨팅[1, 2]은 기존의 포그 컴퓨팅(Fog Computing)[3, 4], 클라우드렛(Cloudlet)과 모바일 엣지 컴퓨팅[5]과 구분되어 설명되기도 한다[6]. 그리고 엣지 컴퓨팅의 중요한 3가지의 요소를 엣지, 계산, 그리고 지능으로 구분한다[7]. 따라서 엣지 컴퓨팅은 근본적으로 많은 양의 데이터를 가공하여 클라우드로 전송하고, 데이터를 기반으로 한 지능적 서비스를 제공하는 것이 가능해야 한다.

엣지 컴퓨팅에 대한 최근 연구는 분산화된 클라우드와 저지연 컴퓨팅, 자원에 대한 분산과 데이터 처리[8], 에너지 소비 및 트래픽 관리[9]에 초점이 맞춰져 있다. 분산화된 클라우드와 저지연 컴퓨팅에 대한 연구는 많은 데이터에 대해 클라우드로 데이터를 전달하는 것을 기반으로 기기 - 엣지 노드 - 클라우드에서 엣지 노드에서의 효율적 데이터 전송 방안, 분산 엣지 노드들 간의 연계, 자원 공유 방안 등을 제시한다. 그리고 자원에 대한 분산과 데이터 처리의 경우 네트워크 처리 기반의 엣지가 클라우드렛 기반 저장소가 연계하여 자원을 효율적으로 분산시키고 데이터를 처리해서 클라우드의 초기 데이터 처리 역할의 일부를 수행하기 위한 방안을 제시하고 있다[10]. 에너지 소비와 트래픽 관리 연구는 실제로 모바일 기기, 즉 작지만 어느 정도의 자원과 연산능력을 지닌 기기로 지능화, 데이터 처리 영역을 넘기기 위한 부분에서 파생되어 이 역할을 모바일 기기와 엣지가 연동할 수 있는 방법들을 고려

한다. 이러한 연구들은 그림 1에서 지능형 기기, 스마트 홈, 지능형 차량 영역에서는 활용 가능하며 지능적 기기와 엣지 간의 연동은 보다 광역적인 서비스를 제공할 수도 있는 요소가 될 수 있다. 협업 엣지를 통해 데이터들을 묶어 특정 영역에서의 서비스를 제시하기도 한다[2]. 이러한 접근법도 공동 도메인에서 연관성을 지닌 서비스 제공을 가정으로 내포한다. 포그 컴퓨팅을 위한 도메인에 따른 요구사항 분석[4]의 경우에도 도메인에 대한 요구사항이 상당히 분산되어 있음을 확인할 수 있다.

스마트 팩토리 영역은 자체가 프라이빗 클라우드(Private Cloud)를 포함하고자 하며, 지역, 빌딩, 심지어 각 셀(Cell)에 따라 다른 센서와 기기들을 다루기도 한다. 그리고 이는 화재 등과 같이 특정 데이터를 통해 유사한 결과가 도출되어지는 것도 있을 수 있으나 대체로 다른 요구와 서비스를 위한 데이터들이 생산되어지기 때문에 독립적 역할을 수행해야 하는 경우가 많다. 또한 끊임없이 생산되는 데이터에 대한 처리와 분석, 학습을 모두 빅데이터 기반의 클라우드에 의존적으로 구현하는 것은 상당히 비효율적이며, 클라우드가 이중의 센서나 기기들에 대한 특성들을 모두 관리해야 하는 문제점들도 존재한다.

III. 학습 팩터 공유 기반의 엣지 컴퓨팅

3.1. 스마트 팩토리의 요구사항

공장은 각 동, 층, 셀 단위로 영역을 구분 한다. 또한 각 영역에 따라 고유의 특성을 지닌 사용자 및 기기를 포함하게 되며, 각 영역별로 게이트웨이 또는 노드를 지니게 된다. 때로는 엔드 포인트(End point)가 다수의 게이트웨이에 연결되기도 한다. 이러한 특성들로 인해 스마트 팩토리는 특수한 요구사항들을 지니게 된다. 그림 2에서와 같이 팩토리 내에 프라이빗 클라우드를 구축하고 모든 데이터를 수집, 가공하여 모델링을 수행한다. 이에 따라 엣지 노드는 공장 내에 들어가는 기기, 센서, 설비와의 연결은 물론 클라우드와 연결을 수행해야 한다. 이러한 구성은 산업에서의 기계설비, 센서와 엣지 간 연결에서의 특성화된 통신, Modbus와 같은, 온전히 엣지 노드의 역할로 구성이 된다. 따라서 일반적인 사물인터넷 플랫폼과 엔드 포인트 연결에서 중간에 엣

지가 연결되어짐으로써 기존 사용되는 통신 프로토콜 (HTTP, MQPP, AMQP 등) 외에 추가적인 규격을 고려해야 함을 의미한다. 이와 마찬가지로 일반적으로 사용하는 데이터 형식(Data(Payload) Format)에 대해서도 단순히 XML/Json, offset, TLV(Type, Length, Value)과 같은 것 뿐 아니라 산업화 기반 또는 자체 규격화한 형식을 사용하는 것도 고려해야 한다. 이러한 요구사항은 데이터 파싱과 필터링 등 전처리 전반에 걸친 상당한 기능적 구현을 필요로 하게 된다.

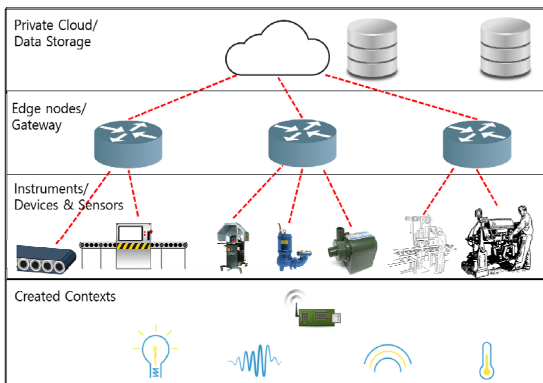


Fig. 2 Edge Computing network architecture in a Factory

이러한 요구사항들을 분류하면 다음과 같다.

1) 감소된 네트워크 대역

엣지 노드로 전송되는 센서 정보는 각 센서별 특성에 따라 다르다. 진동 센서의 경우 x, y, z 축 변동에 대한 정보를 주파수에 따라 생성하게 되는데, 이를 매순간 전송하는 것은 의미가 없을뿐더러 네트워크 자원 소모가 크다. 또한 이를 단순히 모아서 전송을 하고자 하는 경우에도 너무 과도한 양의 데이터를 전송해야 하게 된다. 그렇다고 이러한 데이터를 임의적으로 일정 시간 단위로 평균을 내거나 데이터를 임의로 무작위 추출을 통해 전송하는 것은 학습을 통한 의미성 추출에 문제가 발생할 수도 있다. 특히 기계 학습을 통해 이상치를 추출하고 이러한 기준 값을 기반으로 이상 탐지를 수행하고자 하는 경우 표본의 수나 구성이 상당히 큰 영향을 미치기 때문에 단순 필터링을 수행할 수는 없다. 따라서 데이터의 양은 줄어되, 변이 값이나 줄어든 값의 의미가 클라우드로 전달이 가능해야 한

다. 또한 데이터의 의미성은 보존하되 형식 변형을 통해 데이터를 축소시킴으로써 자원 소모 감소를 고려할 수 있어야 한다.

2) 확장된 분석 지원

팩토리에 구축되는 게이트웨이와 같은 노드는 하드웨어 자원이 충분하지 않기 때문에 강력한 학습 엔진, 인공지능 알고리즘을 수행하기 어렵다. 엄청난 데이터 처리 성능이 필요한 군집화, 분류, 회귀를 수행하기 어려울 뿐 아니라 시스템 자원에 큰 의존성을 지니는 Spark와 같은 프레임워크도, GPU에 영향도가 큰 신경망(Neural Network) 구축에도 어려움이 존재한다. 그럼에도 데이터에 대한 선처리와 효율적 센서, 기기 제어 및 관리를 위한 한정적 데이터 분석과 추론 기능을 요구한다. 이는 센서 노드가 죽거나 엣지와 클라우드 간 연결이 어려운 상황에서도 센서에 대한 재기동이나 상황을 관리자에게 알릴 수 있는 정책 또는 룰이 반영되어 있어야 함을 의미한다.

3) 지역적 룰 관리 제공

산업 환경에서는 특정 장치의 온도가 일정 이상으로 올라가면 알림을 요구한다. 이러한 룰을 구축하고자 하는 경우에 각각의 기기나 센서마다 하나씩 룰을 생성해서는 안 된다. 그러나 현실적으로 IFTTT(If This Then That)와 같은 서비스 플랫폼은 클라우드와 기기가 독립적으로 연결이 되기 때문에, 하나의 룰을 생성할 때는 하나의 장치의 하나의 매개변수 변경을 의미한다. 물론 룰의 Trigger 또는 Condition과 Action을 생성할 때 ‘&’ 조건이 가능하다면, 다수의 장치나 변수를 제어할 수는 있겠지만, 그렇더라도 하나씩 지정을 해야 하는 것이다. 따라서 적어도 하나의 엣지 단위, 또는 지역 단위에 대한 구분이 가능해야 하며, 기계 설비나 센서에 대한 특성에 따른 분류도 필요하다. 이는 조건과 동작 설정이 기기가 아니라 기기의 특성을 지정하는 것도 가능해야 함을 의미한다.

4) 자원 효율성

엣지 노드는 센서의 특성에 따라 때로는 데이터를 스트리밍에 가까운 형태로 수신한다. 또한 일정 간격, 또는 데이터 수집 량에 따라 클라우드로 전송을 해야 한다. 이는 송수신 기능에서의 자원 소모가 큰 것을 의

미하므로, 다른 기능적 요구사항들은 가능하다면 자원 소모에 대한 효율성을 우선적으로 고려한 설계가 이루어져야 한다. 앞선 요구사항 1, 2번에서와 같이 효율적 데이터 전처리와 전송 방안이 필수적인 것이다. 그리고 이러한 센서로부터의 데이터 수신 주기를 고려하여 내부 처리 동작 프로세스에 대한 스케줄 관리가 필요하다.

5) 데이터 기반 처리

사물인터넷의 가장 중요한 요소는 데이터를 수집하는 것이다. 사물인터넷 표준을 지향하는 oneM2M의 경우에도 resource tree를 중심으로 모든 API와 연결성, 관리 방법을 구성한다. 빅데이터나 머신 러닝을 통한 데이터의 분석, 분류, 추론, 학습 기능들은 모두 데이터를 어떻게 보고 판단하느냐의 관점이다. 따라서 엣지 노드는 제한된 리소스를 최대한 활용하여 데이터를 관리할 수 있는 방안이 필요하다. 블록체인과 같이 데이터에 대한 이력을 남기고 관리할 수 있다면 좋겠으나, 수없이 많은 데이터를 모두 저장하고 무결성을 보장하는 것은 어려움이 많다. 따라서 유실 데이터에 대한 복원 방안, 데이터 근집화, 데이터에 기반한 엔드 포인트에 대한 분류, 모델을 제공해야 한다. 그리고 데이터 중 학습 또는 분석에 영향도가 낮거나 의미가 없다 판단되는 인자의 경우 없애거나 줄임으로써 데이터의 효율성을 높이는 방안이 마련되어야 한다.

3.2. 엣지 컴퓨팅 요구사항

요구사항을 기반으로 엣지 노드의 소프트웨어 아키텍처를 설계하면 다음의 그림 3과 같다.

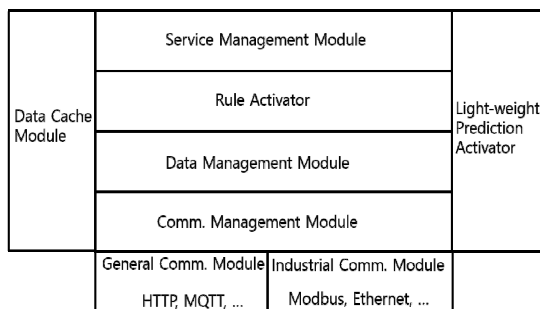


Fig. 3 Edge Computing architecture

앞서 언급했듯이 산업에서는 독자적인 통신 프로토콜을 활용하는 경우가 많다. 따라서 엣지 컴퓨팅은 이러한 통신 프로토콜에 대한 확장성이 유연해야 한다. 필요에 따라 Plug-in이 가능하도록 구현해야하기 때문에 통신 관리 모듈(Communication Management Module)이 필요하다. 이 모듈은 센서로부터 데이터를 수신 가능하도록 네트워크 연결을 관리하며, 이 세션에 대한 운영을 수행한다. 또한 센서가 주기적 전송이 이루어지지 않는 경우 wake-up 또는 상태 확인 메시지를 전송할 수 있어야 한다. 엣지 노드는 이 모듈을 통해 클라우드와 연결하여 데이터를 전송하기도 하고 클라우드로부터 기기 제어 명령은 물론 엣지 노드에서 필요한 정보를 수신할 수 있어야 한다.

본 논문은 엣지 노드와 클라우드 간 네트워크 자원 효율성을 높이기 위해 학습 팩터 공유 방안을 제시한다. 실제로 빅 데이터에서 데이터를 분석할 때 선처리를 수행한다. 데이터가 빈 부분에 대해 어떤 값으로 표기를 할 것인가를 결정해야 한다. 단순히 평균값이나 0 값을 반영하지 않는다. 데이터의 추이를 고려하여 값을 지정하게 된다. 또한 데이터에 대한 이상치를 추출하고 이를 기반으로 데이터의 범위를 산정하게 된다. 데이터들을 근집화하여 각 클러스터에 대한 대표점을 찾기도 하고 분류를 수행하여 무수한 범위 값을 지닌 요소들을 한정된 레벨로 매칭할 수 있다. 데이터 관리와 Instant decision-making을 위해 클라우드로부터 엣지 노드가 학습된 모델을 모두 받을 필요는 없다. 이 모델을 받는다는 것은 동일한 시스템을 보유한다는 것을 의미하지만, 엣지 노드가 동일한 시스템을 구축할 수는 없음을 뜻하는 것이다. 물론 엣지 노드가 이러한 동작을 위해 데이터를 쌓고 경량화 학습 프레임워크를 내포하고 있을 수는 있다. 하지만 이보다는 클라우드에서 추출된 요소를 공유하는 것이 효율적이며 이 값은 온톨로지의 표현 방식에서의 triple과 같이 Subject-Predicate-Object와 같은 표현이 가능하다. 이러한 Triple 값과 룰을 통해 앞서의 센서 데이터의 빈 부분을 채우고 잡음을 제거하는 등의 처리를 하고자 한다. 이에 대한 처리는 다음 표 1과 같다.

Table. 1 Triple Structure and Rule Set

Subject	Predicate	Object
A sensor	locatedIn	a Region
A sensor	contains	temperature A
temperature A	has	B range
B range	fulfill	25
temperature A	has	C range
Extemperature	remove	value

Rule:
 if (Subject[sensor.parameter].contains (range_value) && input[B-1] == Object [sensor.parameter[range_value]]) then input[B] == sensor.parameter[range_value]

 if Subject[sensor.parameter].contains(C_value) && input[B] == Object [sensor.parameter[C_value]] then input[B].remove

이와 마찬가지로 분류나 분석에 대한 결과 역시 형식화된 값의 나열로 전송이 가능하다. 그리고 클라우드에서 룰이 추출되거나 예측을 위한 값이 생성되었다면 이 역시 이와 같은 형태로 전송을 통해 반영이 가능하다. 스마트 팩토리에서 엣지 노드가 수행하는 룰은 데이터 처리 룰과 알림 서비스 룰, 기기 제어 룰로 한정되기 때문에 예측된 룰을 구축하고 인자 값을 변경하는 것으로 구현이 가능하다.

기존의 연구들은 BRMS(Business Rule Management System)을 통해 보다 많은 룰을 하나의 시스템이 효율적으로 관리할 수 있는 것에 초점이 맞춰져 있었다. 그러나 IFTTT 개념이 활성화되면서 이에 대한 연구가 급격히 늘어나고 있다. 이와 유사하게 서버-클라이언트 간 룰을 전송함으로써 클라이언트가 룰 엔진을 고스란히 가져야 하는 부담을 감소시키도록 구현되기도 한다 [11]. 본 논문에서도 이러한 개념을 도입하되 스마트 팩토리에서는 행동 패턴이나 특성이 내부적으로 정형화되기 때문에 룰의 타입과 요소의 전달만으로도 구성이 가능하도록 한다.

데이터 관리 모듈(Data Management Module)은 전송된 요소를 기반으로 데이터에 대한 전처리를 수행한다. 데이터에 대한 클리닝, 통합, 정규화를 통해 데이터를 1차 가공을 수행한다. 요소를 수신하여 처리하기 때문에 이에 대한 자원 소모는 상당히 감소한 상태로 처리가 가능하다. 또한 이 모듈은 유사도가 높아 분석 결과에 영향도가 크지 않은 요소에 대한 값 전체를 제거하거나

범위 외의 특정 값만 남겨두도록 처리하는 역할도 수행할 수 있다.

룰 활성화 모듈(Rule Activator)은 실제 룰 엔진을 모두 포함하지 않는다. 룰 엔진의 경우 룰을 구성하는 형식에 대한 생성과 파싱, 룰 검색, 충돌 관리 등 복잡한 처리가 존재한다. 그러나 여기서의 룰 활성화 모듈은 단순히 룰을 활성화/비활성화하고 인자의 값 변경, 그리고 조건 검색 후 결과 처리 역할만 담당하면 된다. 룰의 변경해야 하는 인자 값은 클라우드를 통해 수신하게 될 것이며, IFTTT의 형태로 간략화된 활성화 역할을 수행하기 때문에 자원의 소모도 높지 않은 상태가 된다 [12, 13].

서비스 관리 모듈(Service Management Module)은 엣지 노드에 부여된 동작들을 수행 가능하도록 한다. 알림의 경우 기본적으로 해당 동작을 클라우드로 명령 전송을 할 수도 있다. 또는 필요한 경우 자체적으로 SMS 또는 E-mail을 통해 메시지를 전달할 수도 있다. 이러한 처리 로직과 액션에 대한 동작 기능을 서비스로 명시한다. 이 모듈은 룰의 결과에 따라 적합한 동작이 이루어지도록 하며, 클라우드를 포함한 외부 다른 시스템과의 연동을 통한 동작 기능도 포함한다.

엣지 컴퓨팅은 시스템의 성능이나 기능의 범위에 따라 데이터의 저장, 가공 범위를 산정할 수 있어야 한다. SQL 데이터 베이스를 가지고 일정 범위의 데이터를 모두 저장하며 처리가 완료된 경우 제거하는 정책을 수립하여 구축하거나 데이터 가공을 위한 최소한의 주기를 캐시에 담고 데이터를 클라우드로 전송하면 캐시를 리프레시하는 형태로 설계할 수도 있다. 그러나 학습 요소(Learning Factor)를 수신하여 처리하기 때문에 많은 데이터를 엣지에서 가져가야할 필요가 없으므로 데이터 캐시 모듈(Data Cache Module)을 구축하고 데이터 관리 정책을 적용한다.

경량화 예측 활성화 모듈(Light-weight Prediction Activator)은 Instant decision-making 동작을 가능하게 한다. 이는 룰과 상호보완적으로 처리가 가능하다. 갑자기 범위 밖의 값이 입력되어질 수 있다. 이 한 번이 실제로 이상이 발생한 것인지, 아니면 단순 센서의 잡음 값인지 판별이 필요하다. n 번의 입력동안은 대기 후 룰을 기동하게 한다거나 이 입력 값과 같은 군집 또는 그룹에 속하는 값들을 비교하여 판별을 하는 등의 정책적이며 학습에 기반한 추론 동작이 필요한 것을 포함하고

록 한다. 이 추론은 이미 클라우드에서 결정된 값을 기반으로 하며, 룰에 대한 제어도 담당하도록 한다.

본 논문에서는 실제 학습이나 룰 관리 기능을 제거한 활성화 모듈 기반의 지능형 엣지 컴퓨팅 아키텍처를 제안한다. 또한 학습 요소를 클라우드와 연계함으로써 이 구조가 가능하도록 하며, Instant decision-making을 지원하도록 한다. 룰에 대한 가공을 편리하게 함으로써, 그리고 학습 요소와 센서와 기기에 대한 컨텍스트 정보를 보유함으로써 룰에 대한 범위적 설정이나 클래스 기반의 설정도 가능하도록 한다. 이러한 프로세스는 자원이 제약된 엣지 노드에서 적절하게 활용이 가능할 것이며, 네트워크 자원과 하드웨어 자원 역시 과부하나 병목을 발생시키지 않도록 구축이 가능하게 한다.

3.3. 엣지 컴퓨팅 구현

모든 지능형 처리를 클라우드가 담당을 하는 경우도 존재한다. Amazon AWS, IBM Watson, IFTTT.com의 경우 클라우드에서 데이터를 수집하고 저장하고 이를 통해 학습을 하거나 사용자로부터 Rule을 받아서 동작이 가능하도록 구현한다. 또한 현재 사물인터넷과 인공지능을 적용하고 있는 공장들에서 도입하고 있는 것 또한 프라이빗 클라우드가 모든 데이터를 처리하는 것이다. 그러나 여기서 발생하는 클라우드에서의 부하로 인해 엣지 도입을 필요로 하고 있다. 그러나 게이트웨이가 룰을 클라우드로부터 받아서 처리하고자 하는 경우 룰의 전송에서 많은 데이터 소비가 필요하게 된다. 엣지에서 최소한의 지능화를 지니기 위해서 룰 엔진까지는 아니어도 룰 실행 모듈 또는 활성화 모듈로써의 기

능은 포함해야 한다.

그림 4에서는 각 룰 활성화 모듈의 특성에 따른 패킷 크기를 보여주며, 이는 HTTP에서 헤더를 제외한 요청 내용의 평균 크기를 나타내며, 동일한 형식의 룰로 비교를 한 것이다. 룰은 단순히 센서 두 개가 각각 특정 임계치를 넘은 값이 들어오면 알람을 요청하는 것으로 수행하였다. Drools는 BRMS으로써 대용량의 룰을 처리할 수 있으며, 동적으로 룰을 관리할 수 있어 많이 활용이 되고 있다.

또한 룰 처리 알고리즘을 내장하고 있어 성능적인 측면에서는 큰 장점을 지닌다. Atooma는 임베디드 IFTTT를 지향하며, 클라우드와 임베디드 기기 간 룰 추천이나 관리를 가능케 하는 시스템을 구축하고 있다. 이 세 가지를 비교한 것은 성능이 좋은 일반적인 룰 엔진의 모델과 IFTTT 기반의 경량화 룰 엔진 모델과 제안한 모델의 각 특성에 따른 비교를 하고자 함이다. 룰을 초기 구조화하는 부분에서는 다른 룰 엔진의 경우 내부적으로 룰 기술 방법을 지니고 있고, 모델도 내부에서 클래스와 같은 형태로 구현되어 있기 때문에 따로 패킷을 전송하지 않는다.

그러나 본 논문에서 제안하는 모델은 앞서 설명한 것과 같이 사전 정의된 룰 구조를 지니고 있되, 매개변수들 간의 관계는 필요시 클라우드로부터 수집을 하게 됨으로써 초기 구조화에서 패킷 전송이 발생한다. 그러나 실제로 이는 초기화 부분이므로 큰 영향을 지니는 지 않는다. Drools의 경우 'rule-when-then-end'와 같은 자체적인 룰 서술 방식을 지니고 있다. 이 규격에 맞춰서 전송을 해야 한다. 또한 Atooma의 경우 XML로 룰을 표현하기 때문에 불필요한 태그가 많이 붙게 되어 단순 조건과 행동의 나열임에도 불구하고 크기가 크다. 그러나 제안한 엣지는 매개변수 타입과 값 세트로 전달이 가능하고 이는 마치 Json으로 표현하듯 그 크기를 줄일 수 있다. 이를 통해 다른 모델들에 비해 약 10%의 크기로도 룰을 전달하고 업데이트할 수가 있다.

서비스 응답 시간을 평가하기 위해 가상 실험 센서를 구성하여 일정 시간마다 센서 값을 전송하도록 하였다. 그리고 이 값은 일정한 범위 내에서 동작을 하다가 일부의 값이 전송하도록 한다. 이 상황에서 룰에 의해 확인하는 것이 아닌 학습된 모델로부터 이상 값을 인지하고 결과를 전송하는데 까지 걸리는 시간을 측정하도록 진행하였다. 클라우드는 컴퓨터 상에서 데이터

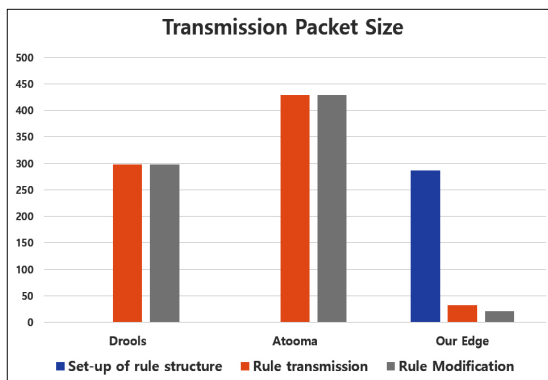


Fig. 4 Transmission Packet Size from Cluster to Edge

를 전송받아서 카산드라(Cassandra)에 저장을 수행하고 이를 Spark에서 클러스터링을 위해 kMeans 알고리즘과 이상 감지를 위한 Anomaly Detection 알고리즘을 활용하였다.

엣지의 경우 JVM 상에서 동작이 가능하도록 한 저사양 노트북을 기준으로 했으며, 이러한 성능을 고려하여 학습을 위해 Smile 라이브러리를 적용하여 분류 수행을 하도록 하였다. 이 클래스는 클라우드로부터 수신하여 모델에 적용하도록 하였으므로, 실제 학습을 수행하기 보다는 클래스를 기준으로 결과를 추출하는 형태를 갖도록 하였다.

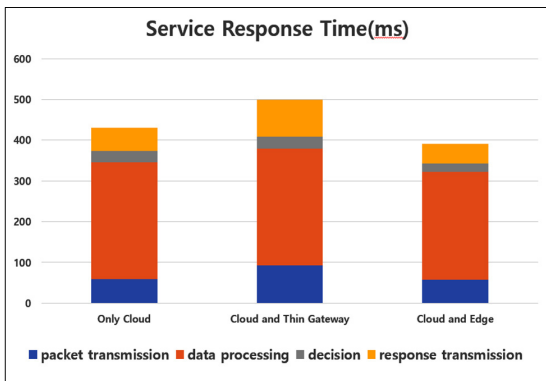


Fig. 5 Service response time with and without Edge scheme

그림 5에서와 같이 제안한 엣지의 구조가 응답 시간과 결정(추론) 시간에서 약 4% 정도의 빠르게 처리가 되는 것을 확인할 수 있다. 기본적으로는 게이트웨이를 거침으로써 발생하는 딜레이가 존재하나, 학습 요소들과 클래스들을 적용함으로써 게이트웨이에서 간단한 이상 상황 감지나 처리가 가능해짐으로써 클라우드를 거치지 않고 결과 추정을 제공할 수 있음으로써 발생하는 처리 성능이 결과로 나타나고 있다. 물론 이는 대용량 센서 데이터의 처리나 이상 상황 또는 잡음 발생이 적은 곳에서는 비효율적일 수 있다. 또한 데이터가 선 처리 동작이 크지 않을 만큼 클린한 경우에도 비효율적일 수 있다. 그러나 스마트 팩토리 영역에서는 유실되는 데이터도, 센서나 주변 상황에 따라 잡음이 발생하는 경우도 많아서 이런 특성에 의해 본 논문에서 제안하는 구조의 효율성을 찾을 수 있다.

IV. 결론

산업 사물인터넷, 스마트 팩토리로 대표되는 4차 산업혁명은 이미 급격하게 확산되고 있다. 이미 GE는 산업 사물인터넷 표준을 통해 기기 및 센서, 환경 정보를 수집하여 각 시설, 설비에 대한 수명이나 상태를 관리하고 있다. 또한 사전 진단을 통해 보다 효율적인 운영이 가능하도록 하고 있다. 이러한 데이터는 블록체인과 같은 유통에서의 운영 기술과 연계되어 부품이나 설비 수리 등에서도 활용하고 있다. 본 논문은 사물인터넷 활용성이 높으나 실제 도메인 특성에 따라 차별화된 진입 방법이 필요한 스마트 팩토리를 분석하였다. 또한 현재 사물인터넷 플랫폼은 급격히 늘어나고 이러한 일반적 산업인터넷 플랫폼을 산업에 적용하려 하면서 발생하는 요구사항에 대한 결핍 문제를 해결하기 위한 연구를 수행하였다. 산업의 특성과 엣지 노드로서의 역할, 한정적 자원에서의 효율적 동작을 위해 엣지 컴퓨팅을 도입은 필수적인 요소로 인식되고 있다. 논문에서 제시한 방안은 성능적 관점에서의 요구사항을 기반으로 하고 있다. 그러나 팩토리는 데이터에 대한 보안과 시스템의 안정성 또한 중요한 요구사항으로 도출된다. 앞으로 이러한 보안 측면에 대한 연구와 산업 특성에 따른 데이터 유실에 대한 무결성 측면에서의 관리 기법 연구가 필요할 것으로 보인다.

REFERENCES

[1] P. G. Lopez, A. Montresor, D. Epema, et al., "Edge-centric Computing: Vision and Challenges," *ACM SIGCOMM Computer Communication Review*, vol. 45, no. 5, pp. 37-42, Oct. 2015.

[2] W. Shi, J. Cao, Q. Zhang, et al., "Edge Computing: Vision and Challenges," *IEEE Internet of Things Journal*, vol. 3, no. 5, Oct. 2016.

[3] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog Computing and its role in the Internet of Things," in *Proc. 1st Edition MCC Workshop Mobile Cloud Computing*, pp. 13-16, Aug. 2012.

[4] C. C. Byers, "Architectural Imperatives for Fog Computing: Use Cases, Requirements, and Architectural Techniques for For-Enabled IoT Networks," *IEEE Communications*

- Magazine*, vol. 55, no. 8, pp. 14-20, Aug. 2017.
- [5] X. Sun, N. Ansari, "EdgeIoT: Mobile Edge Computing for the Internet of Things," *IEEE Communications Magazine*, vol. 54, no. 12, pp. 22-29, Dec. 2016.
- [6] K. Dolui, and S. K. Datta, "Comparison of Edge Computing Implementations: For Computing, Cloudlet and Mobile Edge Computing," *Global Internet of Things Summit (GIoTS)*, Jun. 2017.
- [7] T. Yaofeng, D. Zhenjiang, Y. Hongzhang, "Key Technologies and Application of Edge Computing," *ZTE Communications*, vol. 15, no. 2, pp. 26-34, Apr. 2017.
- [8] A. Houmandadr, S. A. Zonouz, and R. Berthier, "A Cloud-based Intrusion Detection and Response System for Mobile Phones," *IEEE/IFIP International Conference on Dependable Systems and Networks Workshops*, pp. 31-32, Jun. 2011.
- [9] A. Beloglazov, J. Abawajy, and R. Buyya, "Energy-aware Resource Allocation Heuristics for Efficient Management of Data Centers for Cloud Computing," *Future Generation Computer Systems*, vol. 28, no. 5, pp. 755-768, May 2012.
- [10] S. Yang, "IoT Stream Processing and Analytics in the Fsog," *IEEE Communications Magazine*, vol. 55, no. 8, pp. 21-27, Aug. 2017.
- [11] M. Lee, Y. Uhm, Y. Kim, et al., "Intelligent Power Management Device With Middleware Based Living Pattern Learning for Power Reduction," *IEEE Transactions on Consumer Electronics*, vol. 55, no. 4, pp. 2081-2089, Nov. 2009.
- [12] J. C. Na, G. P. Kumar, "Quality of Service in Meta Cloud," *Asia-pacific Journal of Convergent Research Interchange*, vol.1, no.3, pp. 53-57, September 2015.
- [13] J. P. Hong, E. J. Kim, and H. Y. Park, "An analysis of determinants for artificial intelligence industry competitiveness," *Journal of the Korea Institute of Information and Communication Engineering*, vol.21, no.4, pp.663-671, Apr. 2017.



황지온(Zi-on Hwang)

2005년 중앙대학교 전자전기공학부 공학석사
 2011년 중앙대학교 전자전기공학부 공학박사
 2011년 중앙대학교 전자전기공학부 연구교수
 2012년 ~ 현재: 협성대학교 스마트시스템소프트웨어공학과 조교수
 ※관심분야 : IoT, 인공지능, 지식서비스, 유비쿼터스 컴퓨팅, 정보보호, SW교육