

오픈소스 플랫폼 기반의 실시간 환자 대기시간 모니터링 시스템 설계

류우석*

A System Design for Real-Time Monitoring of Patient Waiting Time based on Open-Source Platform

Wooseok Ryu*

*Department of Healthcare Management, Catholic University of Pusan, Busan, 46252 Korea

요 약

본 논문에서는 병원에서 환자의 대기시간을 실시간으로 모니터링하기 위한 오픈소스 기반의 시스템을 제안한다. 환자의 위치 데이터를 실시간으로 분석, 처리하기 위한 고성능 스트림 처리 시스템을 비용 효율적으로 구축하기 위해서는 오픈소스 프로젝트를 활용하는 것이 필요하다. 빅데이터를 처리하기 위한 오픈소스 시스템으로 다양한 하둡 서브프로젝트들로 구성된 하둡 에코시스템이 있다. 본 논문에서는 먼저 시스템 요구사항을 정의하고 하둡 에코시스템에서 이를 만족시키기 위한 몇 가지 오픈소스 프로젝트들을 선정한다. 그리고, 선정된 아파치 스파크, 아파치 카프카 등을 이용한 시스템 구조 설계 및 상세 모듈 설계를 제안한다. 제안된 시스템은 기존 시스템과의 연계 및 오픈소스 프로젝트를 통해 구축비용을 절감할 수 있으며, 또한 분산 스트림 처리를 통해 고성능과 안정성을 확보할 수 있다.

ABSTRACT

This paper discusses system for real-time monitoring of patient waiting time in hospitals based on open-source platform. It is necessary to make use of open-source projects to develop a high-performance stream processing system, which analyzes and processes stream data in real time, with less cost. The Hadoop ecosystem is a well-known big data processing platform consisting of numerous open-source subprojects. This paper first defines several requirements for the monitoring system, and selects a few projects from the Hadoop ecosystem that are suited to meet the requirements. Then, the paper proposes system architecture and a detailed module design using Apache Spark, Apache Kafka, and so on. The proposed system can reduce development costs by using open-source projects and by acquiring data from legacy hospital information system. High-performance and fault-tolerance of the system can also be achieved through distributed processing.

키워드 : 빅데이터, 오픈소스, 환자대기시간, 스파크

Key word : Big Data, Open-Source, Patient Waiting Time, Spark.

Received 8 February 2018, Revised 12 February 2018, Accepted 23 February 2018

* Corresponding Author Wooseok Ryu (E-mail:wsryu@cup.ac.kr, Tel:+82-51-510-0611)

Department of Healthcare Management, Catholic University of Pusan, Busan, 46252 Korea

* 이 논문은 “The Fourth Asia Workshop on IT Convergence of KIIICE 2018”에서 발표된 “Real-Time Monitoring of Patient Waiting Time using Apache Spark” 논문을 확장한 논문임

Open Access <http://doi.org/10.6109/jkiice.2018.22.4.575>

pISSN:2234-4772

©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

병원에서 환자의 대기시간 관리는 병원에서의 환자의 만족도 측면은 물론 의료의 질 관점에서 매우 중요한 요소이다 [1]. 무선통신, IoT(Internet of Things), 빅데이터 [2] 등 정보기술의 급격한 발전에 따라 개별 환자에 대한 위치 확인이 실시간으로 가능해지고 있으며 이에 환자 각각에 대한 대기 시간을 최적화 할 수 있는 기술적 여건이 조성되고 있다. 대량의 환자 데이터를 이용하여 실시간으로 데이터를 분석하고 환자의 대기시간을 모니터링하기 위해서는 대량의 스트림 데이터를 처리하기 위한 고성능 스트림 처리 프로세싱 플랫폼이 요구된다. 하지만, 이러한 스트림 처리 플랫폼은 고성능 하드웨어를 요구하고 있으며 시스템의 도입 비용이 매우 높은 문제가 있다.

본 논문에서는 중소 규모의 병원에서 환자 대기시간을 모니터링하기 위한 비용 효율적인 접근 방법을 논의하고자 한다. 이를 위해 본 논문에서는 오픈소스 기반의 빅데이터 분석 플랫폼인 하둡 에코 시스템을 활용하여 환자의 대기시간을 모니터링하고 관리하기 위한 시스템을 제안하고자 한다. 하둡 에코 시스템 [3]은 아파치 프로젝트 중의 하나로서 빅데이터를 분산 저장하기 위한 하둡 분산파일시스템(HDFS)와 병렬분산 데이터 처리 프레임워크인 맵리듀스(MapReduce)를 포함하고 있는 하둡 플랫폼 [4]을 기반으로 하여 빅데이터 분석에서 필요한 다양한 종류의 오픈소스 프로젝트로 구성된 시스템을 통칭하는 이름이다. 본 논문의 목적은 오픈소스 프로젝트를 활용하여 적은 비용으로 고성능과 확장성을 지원하는 실시간 모니터링 시스템을 구축하기 위한 설계를 제시하는 것이다. 이를 위해 다양한 프로젝트로 구성된 하둡 에코 시스템 실시간 환자 대기시간 모니터링을 구현에 적합한 오픈소스 프로젝트들을 선정하고, 이를 이용한 실시간 환자 모니터링 시스템을 설계하고자 한다.

본 논문의 구성은 다음과 같다. 먼저, 2장에서는 환자 대기시간 모니터링에 대한 기존의 관련 연구를 분석하고 문제점과 본 연구와의 차이점을 기술한다. 3장에서는 실시간 대기시간 모니터링을 위한 시스템 요구사항을 정의한다. 4장에서는 정의된 요구사항을 만족시키기 위한 하둡 오픈소스 프로젝트를 선정하고 이를 기반으로 한 시스템 아키텍처 설계 및 모듈 설계를 제안한다.

마지막으로 5장에서 결론과 향후 연구를 기술한다.

II. 관련 연구

환자 대기시간 관리와 관련한 기존의 연구로서 개별 환자의 예상 대기시간을 이용하여 진료 순서를 동적으로 변경함으로써 진료 대기시간을 단축시키는 상황인지를 이용한 진료 안내 시스템이 제안되어 있다 [5]. 하지만 이 연구는 대기 환자 수와 평균 진료 시간을 이용하여 예상 대기시간을 산출함에 있어서 그 정확성을 평가할 방법이 제시되어 있지 않다. 대기시간을 단축시키기 위한 다른 연구로서 제약이론(TOC)를 병원 수납업무 프로세스에 도입하여 대기시간을 단축시키기 위한 내부 프로세스 개선 방안 연구 및 퍼지 프로세스 마이닝 기법을 이용하여 방사선과에서의 환자 대기시간을 단축시키기 위한 프로세스 개선 기법에 대한 연구가 제안되어 있다[6][7]. 이 연구들은 실시간 환자 데이터를 이용하지 않음에 따라 예외적으로 발생할 수 있는 특정 환자의 이상 대기시간을 탐지하지 못하는 문제가 있다. 연구 [8]는 특정 질환의 진단에 필요한 시간과 환자의 대기시간의 차이를 비교하여 지연시간을 분석함으로써 환자의 개별 질환에 따른 환자 대기시간 단축의 중요성을 제시하였다.

하둡 에코 시스템을 활용한 연구로서 아파치 스파크와 MongoDB를 활용하여 대기시간을 포함한 주요 병원 지표를 실시간으로 관리할 수 있는 데이터케어(DataCare) 시스템이 제안되었다 [9]. 이 연구는 오픈소스를 활용함으로써 확장성과 고성능을 보장하고 있으나 환자 데이터 수집을 비컨 등의 센서 장비에 한정함으로써 실제 병원에서 활용하기에는 제약이 있는 문제가 있다. 본 논문은 병원에 기 설치되어 있는 시스템에서 데이터를 추출하여 확장성 있는 오픈소스 기반 시스템을 제시하는 측면에서 기존의 연구와 차이가 있다.

III. 시스템 요구사항 정의

이 장에서는 병원에서의 환자 대기시간을 모니터링하고 관리하기 위한 시스템 구조 관점에서의 몇 가지 요구사항을 정의한다. 첫 번째 요구사항은 고성능 스트

립 처리이다. 환자 대기시간을 모니터링하기 위해서는 환자가 병원에 도착할 때부터 병원을 나갈 때까지 환자의 위치, 이동과 관련된 다양한 데이터들을 실시간으로 수집하고 이를 분석할 수 있어야 한다. 이 데이터들은 스트림 형태를 띄고 있음에 따라 대량의 스트림 데이터를 실시간으로 처리하는 것이 필요하다.

두 번째 요구사항은 기존의 시스템을 활용한 유연한 데이터 수집이다. 비용 효율적 시스템 구축을 위해서는 기존의 접수/수납(PM/PA), 처방전달시스템(OCS), 전자의무기록(EMR) 등 다양한 병원정보시스템을 활용하여 환자의 개별 위치 관련 데이터를 수집할 수 있어야 한다. 만일 실내위치추적 시스템이 구축되어 있는 경우 라면 비컨, 와이파이, 모바일 디바이스 등을 통해 수집되는 다양한 위치 관련 데이터들을 포함할 수 있으므로 데이터 소스를 유연하게 확장하는 것이 필요하다.

세 번째 요구사항은 환자의 대기시간을 모니터링하기 위해서는 실시간 데이터와 저장 데이터가 결합된 분석이 이루어져야 한다는 것이다. 실시간으로 전송되는 스트림 데이터에서 환자의 대기시간이 지나치게 길거나 혹은 짧은지의 여부를 실시간으로 판단하는 것은 물론, 전송되는 환자의 위치 데이터를 이용하여 환자의 대기 시간 산정 및 분석을 주기적으로 수행하기 위해서는 데이터의 저장, 관리가 동시에 필요하다.

IV. 시스템 설계

4.1. 시스템 아키텍처 설계

이 장에서는 앞에서 제시한 요구사항을 만족시키는 환자 대기시간 모니터링 시스템의 아키텍처 설계를 위해 그림 1과 같이 하둠 오픈소스 프로젝트를 이용한 모니터링 시스템을 제안한다. 본 시스템을 위해서 도입한 오픈소스 프로젝트는 아파치 스파크, 하둠 분산파일시스템(HDFS), 아파치 HBase, 그리고 아파치 카프카이다. 각 오픈소스 프로젝트에 대한 설명은 아래와 같다.

- 아파치 스파크(Apache Spark) : 메인 메모리 기반 고성능 분산 처리 플랫폼으로서 저장 데이터에 대한 배치 형태의 분석은 물론 스파크 스트리밍 라이브러리를 통한 실시간 스트림 처리도 함께 제공하고 있다 [10].

- 아파치 카프카(Apache Kafka) : 분산 메시지 배포 및 관리를 위한 메시징 시스템이다. 발행/구독 모델(subscribe-and-publish)을 분산 환경에서 처리하기 위한 분산 스트리밍 플랫폼이며, 메시지 배포의 안정성을 위해 다수의 서버로 구성된 클러스터 형태로 운용될 수 있다 [11]. 본 논문에서는 아파치 스파크와 연계하여 데이터 소스에서 수집된 다양한 데이터들을 스파크 스트리밍에 스트림 형태로 전송하는 역할을 수행한다.
- HDFS(Hadoop Distributed Filesystem) : 대량의 빅데이터를 분산 저장하기 위한 분산 파일 시스템으로 하둠 플랫폼의 메인 저장소 역할을 수행한다 [4]. 본 논문에서는 스트림 데이터의 배치 분석을 위한 저장소 역할을 수행한다.
- 아파치 HBase(Apache HBase) : NoSQL 중 하나로서 HDFS에 저장된 빅데이터에 대한 랜덤 읽기/쓰기 기능을 지원하는 분산 데이터베이스 시스템이다 [12]. 본 논문에서는 실시간 데이터 분석시 접근될 수 있는 환자의 위치 정보 및 현재 대기 시간 정보들을 저장, 검색하는 역할을 수행한다.

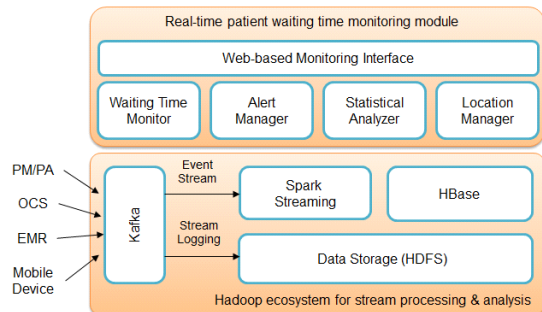


Fig. 1 System architecture for real-time monitoring of patient waiting time

위 오픈소스에 기반한 실시간 환자 대기시간 모니터링 시스템은 크게 분산 메시지 전송 모듈, 데이터 분석 및 스트림 처리 모듈로 구성된다. 두 모듈에 대한 상세 설명은 아래 절에서 기술한다.

4.2. 분산 메시지 전송 모듈

환자의 이동과 관련된 데이터는 다양한 병원정보시스템에 분산되어 저장되어 있다. 예를 들어 환자접수수납(PM/PA) 및 처방전달시스템(OCS)은 환자의 접수가

발생할 때, 환자의 진료가 시작될 때, 진료 후 처방이 발생하였을 때, 그리고 병원비를 수납할 때 등의 다양한 상황이 발생할 때 해당 데이터를 타임스탬프와 함께 데이터베이스에 기록하여 저장한다. 전자진료기록시스템(EMR)은 환자가 의사를 통해서 진료를 받는 동안 해당 진료 기록을 데이터베이스에 저장하고 있으며, 간호정보시스템(NIS), 임상병리시스템(LIS)등도 해당 환자에 대한 기록을 시간정보와 함께 기록하고 있다. 각각의 데이터들은 환자의 대기시간 예측을 위한 중요한 자료로 활용될 수 있다. 하지만 이러한 기존의 병원정보시스템들은 실시간 대기시간 모니터링을 위한 스트리밍 인터페이스를 갖고 있지 못할 수 있다.

분산 메시지 전송 모듈의 목적은 병원정보시스템 내 각종 시스템에 분산되어서 저장되어 있는 환자 관련 데이터들을 수집하고 이를 스트림 형태로 모니터링 모듈에 전송하는 것이다. 아파치 카프카는 본 모듈에서 다양한 병원정보시스템으로부터 환자 이벤트 데이터를 수집하고 이를 스트림 처리 모듈로 전송하는 기능을 수행한다. 본 논문에서 아파치 카프카는 OCS, EMR과 같은 데이터소스 시스템과 데이터 분석 및 스트림 처리 모듈 간의 연계 역할을 수행함으로써 기존 시스템의 데이터베이스로부터 생성되는 데이터들을 스트림 처리 모듈로 스트림 형태로 전송하게 된다. 아파치 카프카를 포함한 분산 메시지 전송 모듈의 구조는 그림 2와 같다.

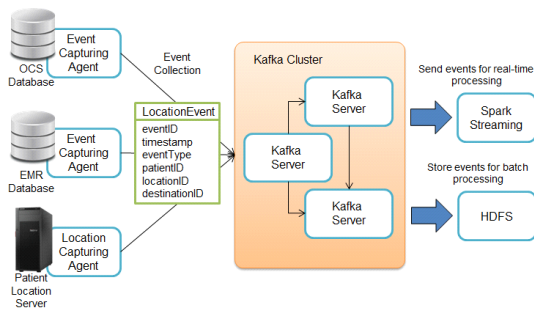


Fig. 2 Distributed message publishing module

그림 2에서 이벤트 캡처링 에이전트(Event Capturing Agent)는 기존의 병원정보시스템을 모니터링하면서 환자의 이벤트 데이터를 수집하고 이를 카프카 클러스터에 전송하는 역할을 수행한다. 이벤트 캡처링 에이전트의 구현은 기존의 시스템에 따라 달라질 수 있다. 만일

병원정보시스템의 데이터베이스에 접근이 가능한 경우 이 에이전트는 환자 이벤트 데이터를 저장하는 특정 테이블의 레코드를 모니터링하면서 해당 레코드가 데이터베이스에 추가될 때 동작하는 트리거(Trigger) 함수 형태로 구현될 수 있으며 이 트리거 함수를 통해 에이전트로 데이터가 전송되면 카프카의 Connector API와 Producer API를 통해서 카프카 클러스터로 데이터를 전송하게 된다.

이벤트 캡처링 에이전트가 카프카 클러스터로 데이터를 전송할 때의 데이터 모델은 표 1과 같이 타임스탬프, 이벤트의 종류, 환자ID, 위치ID, 그리고 목적지ID를 포함한 위치이벤트(LocationEvent)로 정의할 수 있다. 이 이벤트 데이터는 카프카의 내부에서 사용되는 추상적 데이터 타입인 토픽(Topic)형태로 스트림 데이터 처리 모듈로 전송되게 된다.

Table. 1 Field Description of LocationEvent

Field Name	Description
eventID	An identifier of an event
timestamp	a timestamp when the event is created
eventTypeID	An enumerated ID for representing type of the event such as REGISTER and PAYMENT.
patientID	An encrypted identifier of a patient
locationID	An identifier of a location at which the event is generated
destinationID	An identifier of a location where the patient is waiting

카프카 클러스터는 앞 절에서 기술한 바와 같이 메시지 수집/배포 성능 및 안정성 향상을 위해 여러 대의 서버로 구성된 클러스터 형태로 구축될 수 있다. 이벤트 캡처링 에이전트에서 생성되어 카프카로 전송된 위치이벤트 데이터들은 스트림 형태로 두 곳에 전송되는데 하나는 실시간 모니터링을 위한 스트림 처리 모듈이며, 나머지 하나는 배치 분석을 위해 HDFS에 모든 스트림 데이터를 저장한다. 이때 스트림을 소비(Consume)하는 스트림 처리 모듈은 카프카의 Consumer API와 Streams API를 이용하여 구현된다. 본 논문에서는 이벤트 캡처링 에이전트와 카프카 시스템을 구분하여 모듈 구조를 설계함으로써 다양한 데이터 소스로부터의 메시지 수집을 단순화하고 데이터 소스를 쉽게 확장할 수 있다.

또한, 카프카 클러스터를 통해 메시지 전달의 안정성 또한 확보할 수 있다.

4.3. 데이터 분석 및 스트림 처리 모듈

데이터 분석 및 스트림 처리 모듈은 분산 메시지 전송 모듈로부터 수집되는 스트림 데이터들의 분석을 통해 환자의 대기시간을 산출하고 이를 모니터링하여 대기시간 초과, 응급상황 등의 발생 시 이를 탐지하고 사용자에게 알리는 역할을 수행하는 모듈이다. 이 모듈은 그림 3과 같이 크게 두 개의 하위 모듈로 나뉘는데 하나는 저장된 데이터를 분석하는 배치 분석 모듈과 나머지는 실시간 모니터링 모듈이다.

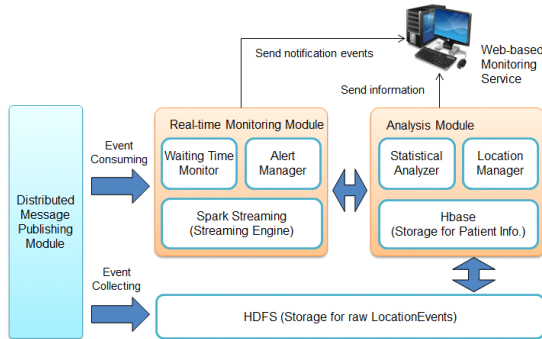


Fig. 3 Data analysis and distributed stream processing using Spark Streaming

데이터 분석 모듈은 위치이벤트 데이터들을 이용하여 환자의 진료대기와 관련한 일별, 요일별 등 각종 상황별 평균 대기시간 및 허용 가능한 대기시간을 시간대 별로 주기적으로 산출하고 이를 저장 관리 하는 역할을 수행한다. 이때의 입력 데이터는 로그 형태로 HDFS에 저장되고 있는 위치이벤트 데이터이며, 산출된 평균 대기시간 데이터는 실시간 모니터링에서 환자의 이벤트 데이터와의 비교 시 빠르게 조회될 수 있도록 HBase에 저장한다.

실시간 모니터링 모듈은 스트림 모니터링을 통해 개별 환자의 위치를 파악하고 해당 위치에서의 현재 대기 시간을 산출한다. 실시간 모니터링 모듈은 아파치 스파크 스트리밍 라이브러리를 이용하는데 스파크 스트리밍은 입력으로 들어오는 위치이벤트 데이터를 RDD로 추상화하여 처리하고 RDD가 연속되는 DStream을 변환하면서 구현한다. 이때 DStream을 변환하는 논리적

흐름은 아래 순서와 같다.

- 스트림에 포함된 위치이벤트 각각에 대해 HBase에 조회를 통해 현재 대기 시간을 산출한다. 해당 환자의 위치가 새로 변경된 경우 현재 대기 시간을 초기화 한다. 산출된 현재 대기 시간은 HBase에 별도 저장한다.
- 환자의 현재 대기 시간을 해당 상황별 평균 대기 시간과 허용 가능한 대기시간과 비교한 결과를 기록한다. 이때의 허용 가능한 대기시간은 기존 대기시간들의 표준편차를 이용하여 산출할 수 있으며 또한 사용자를 통해서도 지정이 가능하다.
- 환자의 현재 대기시간이 허용 가능한 대기시간을 초과하는 이벤트를 필터링한 후 필터된 응급 이벤트 정보를 관리자에게 알릴 수 있도록 알람 관리자(Alert Manager)에게 전송한다.

알람 관리자는 대기시간을 초과한 환자 정보를 관리자에게 전달하여 관리자가 실시간으로 환자의 대기상황을 모니터링하고 이에 대한 조치를 취할 수 있도록 하는 웹 기반의 모니터링 서비스로 구현될 수 있다. 본 시스템에서 스파크 플랫폼을 사용함으로써 배치 분석과 실시간 분석에 모두 대응할 수 있으므로 구현의 편의성은 물론 분산 처리를 통한 고성능 스트림 데이터 처리 능력도 확보할 수 있는 장점이 있다.

V. 결론

본 논문에서는 병원에서 환자의 대기시간을 실시간으로 모니터링하고 이상 탐지 시 이를 보고하기 위한 오픈소스 기반의 실시간 환자 대기시간 모니터링 시스템을 설계하였다. 본 논문의 목적은 기존의 병원정보시스템이 구축되어 있는 병원 환경에서 확장성과 안정성을 보장하는 실시간 환자 모니터링 시스템을 비용 효율적으로 구축하기 위해 오픈소스 플랫폼을 활용한 시스템 구축 방안을 제안하는 것이다. 이를 위하여 먼저 시스템 요구사항을 정의한 후 요구사항에 부합하는 오픈소스 프로젝트를 선정하여 이를 이용한 시스템을 설계하였다. 오픈소스 프로젝트를 기반으로 함에 따라 비교적 적은 비용으로 시스템의 구축이 가능하며 또한 기존

시스템의 변경 없이 모니터링이 가능하다는 것을 설계 관점에서 입증하였다. 본 연구는 시스템 요구분석 및 구조 설계 제시를 통해 시스템 구축 가능성을 제시함에 따라 향후 연구로서 상세한 설계 및 구현을 통해 본 연구의 실현 가능성을 추가로 입증하는 것이 필요하다.

ACKNOWLEDGEMENT

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIP) (No. NRF-2016R1C1B1012364).

REFERENCES

- [1] J. C. Lee, and M. H. Lee, "Big data-based information recommendation system," *Journal of the Korea Institute of Information and Communication Engineering*, vol. 22, no. 3, pp. 443-450, Mar. 2018.
- [2] I. M. Datuk, et al. "Hospital waiting time: the forgotten premise of healthcare service delivery?," *International Journal of Health Care Quality Assurance*, vol. 24, no. 7, pp. 506-522, 2011.
- [3] S. Ha, S. Lee, and K. Lee, "Development of Reference Architecture Based on Big Data Ecosystem," *Journal of Security Engineering*, vol. 11, no. 6, pp.511-522, Dec. 2014.
- [4] Apache Hadoop Official Homepage [Internet]. Available: <http://hadoop.apache.org>.
- [5] H. Y. Jung, J. W. Park, and Y. K. Lee, "A Context-Aware Treatment Guidance System," *Journal of The Korea Society of Computer and Information*, vol. 19, no. 1, pp. 141-147, Jan. 2014.
- [6] C. S. Park, and S. H. Koh, "A Case Study on the Improvement of General Hospital Outpatients Waiting Time using TOC Methodology," *Korea Journal of Hospital Management*, vol. 16, no. 1, pp. 77-100, Mar. 2011.
- [7] S. I. Park, and K. B. Kim, "Analysis of The Delayed Time in Patients with Acute Appendicitis," in *Proceedings of the Korean Institute of Information and Communication Sciences*, pp. 889-892, 2013.
- [8] K. Ganesha, S. Dhanush, and S. M. S. Raj, "An approach to fuzzy process mining to reduce patient waiting time in a hospital," in *Innovations in Information, Embedded and Communication Systems (ICIIECS)*, pp. 1-6, 2017.
- [9] A. Baldominos, F. D. Rada, Y. Saez, "DataCare: Big Data Analytics Solution for Intelligent Healthcare Management," *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 4, no. 7, pp. 13-20, 2018
- [10] M. Zaharia, et al. "Apache spark: a unified engine for big data processing," *Communications of the ACM*, vol. 59, no.11, pp. 56-65, Nov. 2016.
- [11] Apache Kafka [Internet]. Available: <http://kafka.apache.org>.
- [12] A distributed database for large datasets [Internet]. Available: <http://hbase.apache.org>.



류우석(Wooseok Ryu)

1997년 부산대학교 컴퓨터공학과 졸업 (공학사)
1999년 부산대학교 대학원 컴퓨터공학과 졸업(공학석사)
2012년 부산대학교 대학원 컴퓨터공학과 졸업(공학박사)
2013년 ~ 현재 부산가톨릭대학교 병원경영학과 조교수
※ 관심분야: 의료정보, 스트림 데이터 처리, U-Health, 빅데이터, 허둠 플랫폼