한국정보전자통신기술학회논문지(jkiiect)'21-8, Vol.14 No.4
논문 21-14-04-300 http://dx.doi.org/10.17661/jkiiect.2021.14.4.300

방대한 IoT 장치 기반 환경에서 효율적인 빅데이터 수집 기법 설계

최종석*, 신용태**

Design of Efficient Big Data Collection Method based on Mass IoT devices

Jongseok Choi*, Yongtae Shin**

요 약 IT기술의 발달로 인해 최근 IoT 장비에 적용되는 하드웨어 기술이 저비용, 고성능 RF 및 연산장치를 사용한 스마트 시스템들로 변화되고 있다. 그러나 방대한 양의 IoT 장비들이 설치된 인프라 환경에서 빅데이터 수집은 전송되는 데이터간 병목현상으로 인해 수집 서버의 부하가 발생한다. 이로인해 데이터수집 서버로 전송되는 데이터는 패킷 손실 및 데이터 처리율 감소 현상이 발생한다. 따라서 방대한 양의 IoT 장비들이 설치된 인프라 환경에서 효율적인 빅데이터 수집 기법이 필요하다. 이에 본 논문에서는 방대한 양의 IoT 장비들이 설치된 인프라 환경에서 효율적인 빅데이터 수집 기법을 제안한다. 성능평가 결과, 제안하는 기법의 패킷 손실 및 데이터 처리율은 전송되는 파일의 손실없이 전송이 완료된다. 향후 본 설계를 기반으로 시스템이 구현이 필요하다.

Abstract Due to the development of IT technology, hardware technologies applied to IoT equipment have recently been developed, so smart systems using low-cost, high-performance RF and computing devices are being developed. However, in the infrastructure environment where a large amount of IoT devices are installed, big data collection causes a load on the collection server due to a bottleneck between the transmitted data. As a result, data transmitted to the data collection server causes packet loss and reduced data throughput. Therefore, there is a need for an efficient big data collection technique in an infrastructure environment where a large amount of IoT devices are installed. Therefore, in this paper, we propose an efficient big data collection technique in an infrastructure environment where a vast amount of IoT devices are installed. As a result of the performance evaluation, the packet loss and data throughput of the proposed technique are completed without loss of the transmitted file. In the future, the system needs to be implemented based on this design.

Key Words : Big Data, Hadoop, IoT, Smart Factory, Clustering

1. 서론

IT기술을 발달로 인해 최근 IoT 장비에 적용되는 하드웨어 기술들은 저비용, 고성능 RF 및 연산장치를 사용한다. 이로 인해 최근 개발되는 IoT 장비들은 서버와 통신을 위해 IPv6나 IPv4를 사용한다. IoT와 결합된 스마트 시스템들은 IPv6나 IPv4 기술적용으로 인해 각 장비들의 일별 로그 데이터를 수집하여 데이터분석을 통해 장치별 생명주기 및 장애를 예측한다.

그러나 방대한 양의 IoT 장비들이 설치된 인프라 환경에서 빅데이터 수집은 전송되는 데이터간 병목현상으로 인해 수집 서버의 부하가 발생한다. 이로인해 데이터수집 서버로 전송되는 데이터는 패킷 손실 및 데이터 처리율 감소 현상이 발생한다.[1][2]

따라서 방대한 양의 IoT 장비들이 설치된 인프라 환경에서 효율적인 빅데이터 수집 기법이 필요하다.

이에 본 논문에서는 방대한 양의 IoT 장비들이 설치된 인프라 환경에서 효율적인 빅데이터 수집 기법을 제안한다.

*Spartan Software Education Institute, Soongsil University (jschoi@ssu.ac.kr)
**Corresponding Author : School of Computing, Soongsil University (shin@ssu.ac.kr)
Received August 13, 2021 Revised August 14, 2021 Accepted August 20, 2021

제안하는 기법은 빅데이터 수집 단계과 빅데이터 저장 단계로 구성된다. 빅데이터 수집 단계는 방대한 양의 IoT 장비들을 클러스터링 하기 위한 클러스터링 단계와 클러스터링 된 IoT 장비들을 데이터를 전송하기 위한 전송 단계로 구성된다. 빅데이터 저장단계는 빅데이터 수집 단계에서 구성된 클러스터링을 기반으로 클러스터링마다 큐를 생성하고, 생성된 큐는 클러스터링에 속한 IoT 장비들로부터 데이터를 전달받는다. 성능 평가 결과, 제안하는 기법의 패킷 손실 및 데이터 처리율은 전송되는 파일의 손실없이 전송이 완료된다.

본 논문의 구성은 다음과 같다. 2장에서는 플럼 기반의 데이터 흐름을 확인하고, 제안하는 알고리즘에 필요한 요구사항을 도출한다. 3장에서는 본 논문에서 제안하는 알고리즘을 구현하며, 4장에서는 제안한 알고리즘과 성능을 분석한다. 마지막 5장에서는 결론 및 향후 연구 과제를 제시한다.

2. 관련 연구

본 장에서는 가장 보편적인 빅데이터 플랫폼인 하둡의 서브 프로젝트 중 데이터 수집을 위한 플럼에 대해 알아보고 요구사항을 도출한다.

2.1 아파치 플럼

하둡은 하둡에코시스템 구축을 통해 빅데이터 수집, 저장, 처리, 분석, 시각화 등 빅데이터 처리과정을 제공하는 보편적인 빅데이터 플랫폼이다.

아파치 플럼은 여러 서비스를 제공하는 서버에 적재되어 있는 로그들을 하나의 로그 수집 서버로 모으는 역할을 수행하는 빅데이터 수집 기술이다. 아파치 플럼은 주로 파일 단위 빅데이터 수집에 최적화되어 있는 기술이다. 플럼은 스트림 지향의 데이터 흐름(Data Flow)을 기반으로 하며, 지정된 모든 서버로부터 파일들을 수집한 후 하둡의 하둡 분산 파일 시스템과 같은 중앙 스토리지에 적재한다. 아파치 플럼의 데이터 흐름은 스트림 지향의 데이터 흐름을 기반으로 한다. 데이터 흐름은 하나의 데이터 스트림이 생성지에서 목표지로 전달되어 처리되는 방식이다. 데이터 흐름은 이벤트를 전송하고 수집하는 일련의 노드(Node)들로 구성된다.

〈그림 1〉은 아파치 플럼의 데이터 흐름을 나타낸다.[3]

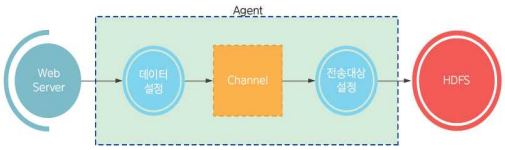


그림 1. 아파치 플럼의 데이터 흐름
Fig. 1. Dataflow in Apache Plum

아파치 플럼의 계층은 에이전트 계층(Agent Tier), 콜렉터 계층(Collector Tier)과 스토리지 계층(Storage Tier) 3가지로 구성된다. 〈그림 2〉는 아파치 플럼의 계층을 나타낸다.

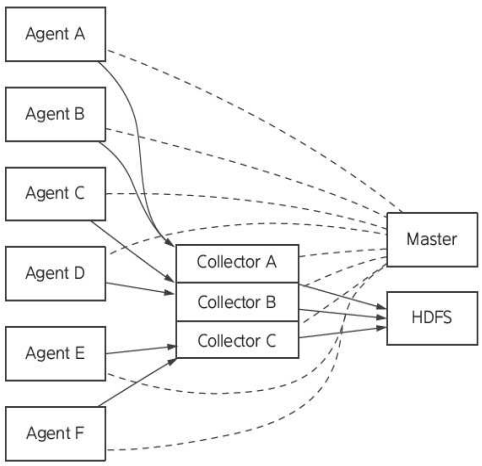


그림 2. 아파치 플럼의 계층
Fig. 2. Layers of Apache Plum

에이전트 계층은 각 에이전트 노드로 구성된 영역이다. 에이전트 계층에서 각 에이전트 노드들은 수집할 로그 데이터가 생성되는 장비에 설치하는 것이 일반적이다. 데이터를 생성하는 장비가 여러 개일 경우의 에이전트 설치는 각 장비마다 에이전트 노드를 설치하고, 이 에이전트 노드들은 에이전트 계층을 형성한다.[4]

콜렉터 계층은 에이전트 노드로부터 받은 정보를 수집하는 영역이다. 에이전트 노드에서 수집한 데이터는 콜렉터 노드(Collector Node)로 전송이 된다. 콜렉터

노드는 보통 다른 장비에 있으며, 여러 콜렉터 노드로 구성할 수 있다. 에이전트 노드에서 콜렉터 노드로 데이터를 전송할 때는 어떤 데이터를 어디로 보내고 어떻게 처리할 것인지 등에 대한 데이터 흐름을 설정할 수 있고, 이 설정대로 데이터를 이동시켜 스토리지 계층에 저장한다.[5][6]

스토리지 계층은 에이전트 노드 및 콜렉터 노드의 설정을 관리하는 마스터 노드(Master Node)와 데이터가 저장되는 하둡 분산 파일 시스템으로 구성된다. 마스터 노드의 주요 역할은 이러한 데이터 흐름을 설정한다. 즉, 마스터 노드는 각 논리적 노드를 프로그램을 통해 설정할 수 있으며, 이 역할은 플럼의 가장 큰 장점 중 하나이다. 플럼은 각 노드를 실행되는 상태에서도 마스터 노드를 통해 자유롭게 설정을 변경할 수 있다. 이는 어디서 로그 데이터를 가져오고 어떻게 처리하고 어디에 저장할 것인지를 동적으로 계속 변경할 수 있다는 것이다.[5][7]

2.2. 요구사항 도출

본 절은 앞서 선행 연구한 빅데이터 수집 기술을 통해 요구사항을 도출한다.

아파치 플럼은 빅데이터 수집 기술의 하나로 파일 및 실시간 빅데이터 수집 기능 제공한다. 특히 아파치 플럼은 여러 서버 및 서비스들로부터 발생되는 파일에 대해 최적화되어 있다.

그러나 IoT 인프라 환경과 같이 방대한 양의 IoT 장비가 설치되는 환경은 아파치 플럼을 적용함에 있어 어려움이 있다. 카프카 또한 유사한 형태로 빅데이터를 수집하는 경우에 사용될 수 있으나, 프로듀서와 컨슈머, 메시징 시스템을 복합적으로 사용하여야 하여, 메시지 수정이 불가피할 경우 성능이 급격히 감소할 수 있다. 다수의 IoT 장비는 지정된 콜렉터 영역를 통해 데이터를 전달한다. 그러나 콜렉터 영역은 자신이 처리할 수 없는 양의 파일이 동시 다발적으로 수집되면 부하가 발생한다. 이로 인해 수집되어야 할 파이들은 패킷 손실 및 전송지연이 발생한다.[8]

따라서 방대한 양의 IoT 장비들이 설치된 인프라 환경에서 효율적인 빅데이터 수집 기법이 필요하다.

3. 제안하는 빅데이터 수집 기법

제안하는 빅데이터 수집 기법은 빅데이터 수집 단계와 빅데이터 저장 단계로 구성된다. 빅데이터 수집 단계는 다수의 IPv6 주소를 가진 다수의 사물인터넷 장비로부터 로그 파일을 수집하는 역할을 수행한다. 빅데이터 저장 단계는 수집된 로그 파일들을 하둡에코시스템에 저장하는 역할을 수행한다.

3.1. 인프라 환경 구성

〈그림 3〉은 제안하는 빅데이터 수집 기법의 구현을 위한 인프라 환경 구성을 나타낸다.

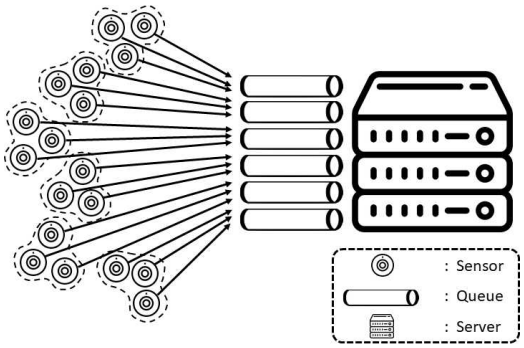


그림 3. 인프라 구성 환경
Fig. 3. Infrastructure configuration environment

인프라 환경 구성은 제안하는 빅데이터 수집 기법의 빅데이터 수집 단계에서 설치된 IoT 장치들을 영역별 클러스터링을 구성한다. 클러스터링 영역이 설정된 IoT 장비들은 자신이 속한 클러스터 영역을 큐로 파일을 전송한다.

3.2. 빅데이터 수집 단계

빅데이터 수집 단계는 클러스터링 단계와 전송 단계로 구성된다. 클러스터링 단계는 방대한 양의 IIoT 장비들을 묶어 클러스터링 영역을 생성하는 역할을 수행한다. 전송 단계는 생성된 클러스터링 영역별 로그 전

송을 위한 에이전트를 설정하고 파일을 전송하는 역할을 수행한다.

3.2.1. 클러스터링 단계

클러스터링 단계는 IoT 장치마다 파일 전송을 위한 큐 설정을 빅데이터 수집 서버와 통신을 수행한다. 빅데이터 수집 서브는 수집할 IoT 장치들의 큐 및 설정 관리를 위한 코디네이터의 역할을 수행한다. IoT 장비마다 사용된 큐 번호는 빅데이터 수집 서버가 큐 정보를 전송하여 지정한다. 〈그림 4〉는 빅데이터 수집 서버에서 IoT 장치에 큐 정보를 전송을 나타낸다.

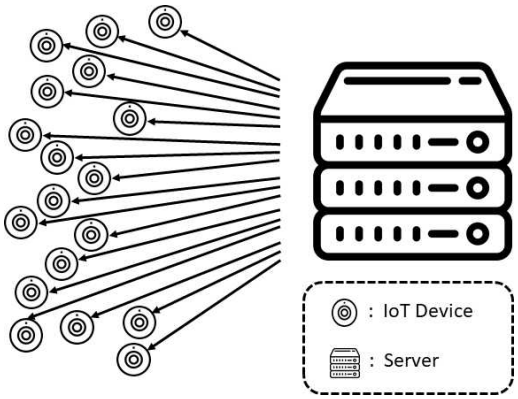


그림 4. 큐 정보 전달 과정
Fig. 4. Queue information transfer process

3.2.2. 전송 단계

전송 단계는 IoT 장치들이 빅데이터 수집 서버로부터 지정된 큐에 데이터를 전송하는 단계이다. 전송되는 데이터는 큐에 저장되며, 도착순서에 맞게 저장되는 FIFO(First In, First Out) 방식으로 사용한다. 〈그림 5〉는 IoT 장치들의 데이터 전송을 나타낸다.

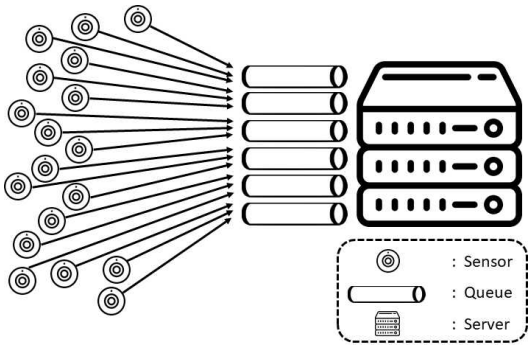


그림 5. 수집된 데이터를 전송하는 전송 단계
Fig. 5. Transmission step to transmit the collected data

3.2. 데이터 저장 단계

데이터 저장 단계는 큐에 저장되는 데이터를 하둡 에코 시스템의 하둡 분산 파일 시스템에 저장하는 역할을 수행한다. 하둡 분산 파일 시스템의 저장 구조는 큐마다 폴더를 생성하여 병렬 저장에 따른 문제를 해결한다.

4. 성능평가

본 장에서는 제안하는 빅데이터 수집 기법의 성능평가를 위해 아파치 플펌과 비교하며, 패킷 손실율과 데이터 처리율을 분석한다.

4.1 실험 환경

〈표 1〉은 제안하는 빅데이터 수집 기법의 성능평가를 위한 실험 환경을 나타낸다.

표 1. 실험 환경
Table 1. Experiment environment

Item	Value
Collect Server	1 Count
IoT Device	10,000 Count
File Size	100mb
Simulation Tool	jMeter

실험환경은 빅데이터 수집을 위한 서버 1대와 파일을 전송하는 IoT 장치는 10,000대로 정의하며, 전송하

는 파일의 크기는 각 100mb한다. 실험을 위한 시뮬레이션 도구는 jMeter를 사용한다. 〈표 2〉는 빅데이터 수집 서버의 스팩을 나타낸다.

표 2. 빅데이터 수집 서버 성능
Table 2. Big data collection server performance

Item	Setting
OS	Ubuntu 18
CPU	8 Core
Ram	16G
Main Disk	256G SSD
Hadoop Version	2.9.2
Hadoop Type	Stand Alone

4.2 패킷 손실율

패킷 손실율은 IoT 장치에서 빅데이터 수집 서버로 전달되는 파일에 대한 패킷들의 손실 양을 계산하여 손실율을 분석한다. 동시 전송하는 IoT 장치의 수는 1,000단위마다 패킷 손실율을 측정하였다.

〈표 3〉〈그림 6〉은 IoT 장치 수에 따른 패킷 손실율 분석을 나타낸다.

표 3. IoT 장치 수에 따른 손실율 분석
Table 3. Loss rate analysis according to the number of IoT devices

IoT Device Count	Pro.	Flume
1,000	0	0.99
2,000	0	1,31
3,000	0	1.51
4,000	0	1.74
5,000	0	1.89
6,000	0	2.04
7,000	0	2.37
8,000	0	2.99
9,000	0	3.77
10,000	0	4.61

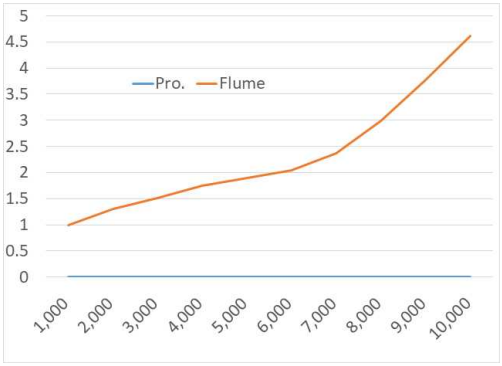


그림 6. IoT 장치 수에 따른 손실율 분석
Fig. 6. Loss rate analysis according to the number of IoT devices

분석 결과, 제안하는 빅데이터 수집 기법의 패킷 손실율은 전혀 발생하지 않았지만, 기존 기법은 동시 전송하는 IoT 장치의 수가 1,000부터 패킷 손실율이 발생한다. 또한 기존 기법은 IoT 장치의 수의 증가될수록 패킷의 손실율이 증가됨을 알 수 있다.

제안하는 빅데이터 수집 기법의 패킷 손실율이 발생하지 않는 이유는 제안하는 빅데이터 수집 기법은 수집되는 장비별 클러스터링을 통해 병목현상을 최소화하며, 큐를 활용하였기 때문이다.

4.3 데이터 처리율

데이터 처리율은 IoT 장치에서 빅데이터 수집 서버로 전달되는 파일에 대해 하둡 분산 파일시스템에 정상적으로 저장된 파일의 수를 계산하여 데이터 처리율을 분석한다. 동시 전송하는 IoT 장치의 수는 1,000단위마다 데이터 처리율을 측정하였다.

〈표 4〉〈그림 7〉은 IoT 장치 수에 따른 데이터 처리율을 나타낸다.

표 4. IoT 장치 수에 따른 데이터 처리율 분석
Table 4. Data throughput analysis according to the number of IoT devices

IoT Device Count	Pro.	Flume
1,000	100%	99.12%
2,000	100%	98.94
3,000	100%	98.34
4,000	100%	97.74
5,000	100%	97.11
6,000	100%	96.88
7,000	100%	96.0
8,000	100%	95.57
9,000	100%	94.91
10,000	100%	94.13

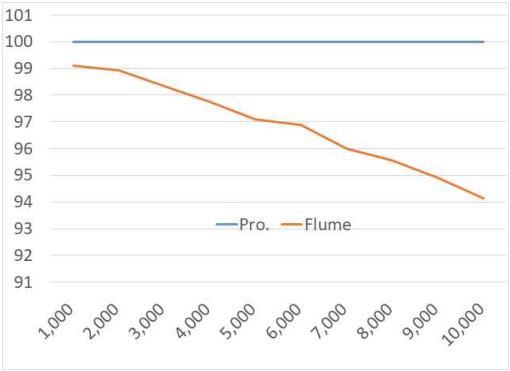


그림 7. IoT 장치 수에 따른 데이터 처리율 분석
Fig. 7. Data throughput analysis according to the number of IoT devices

분석 결과, 〈그림 7〉과 같이 제안하는 빅데이터 수집 기법의 데이터 처리율은 1,000개부터 10,000개 동시 전송된 파일 모두 하둡 분산 파일 시스템에 정상적으로 저장된다. 기존 기법은 동시 전송하는 IoT 장치의 수가 1,000부터 정상적으로 저장되는 못한 파일들이 발생한다. 특히 동시 전송하는 IoT 장치의 수가 10,0000인 경우, 100 중 6개는 저장되지 못하는 현상이 발생한다.

즉, 동시 전송하는 IoT 장치의 수가 증가될수록 데이터 처리율을 급격하게 감소한다.

제안하는 빅데이터 수집 기법의 데이터 처리율은 100%로 모두 정상적으로 저장된 이유는 제안하는 빅데이터 수집 기법은 수집되는 장비별 클러스터링을 통해 병목현상을 최소화하며, 큐를 활용하였기 때문이다.

5. 결론

본 논문에서는 방대한 양의 IoT 장비들이 설치된 인프라 환경에서 효율적인 빅데이터 수집 기법을 제안한다.

제안하는 기법은 빅데이터 수집 단계과 빅데이터 저장 단계로 구성된다. 빅데이터 수집 단계는 방대한 양의 IoT 장비들을 클러스터링 하기 위한 클러스터링 단계와 클러스터링 된 IoT 장비들을 데이터를 전송하기 위한 전송 단계롤 구성된다. 빅데이터 저장단계는 빅데이터 수집 단계에서 구성된 클러스터링을 기반으로 클러스터링마다 큐를 생성하고, 생성된 큐는 클러스터링에 속한 IoT 장비들로부터 데이터를 전달받는다. 성능 평가 결과, 제안하는 기법의 패킷 손실 및 데이터 처리율은 전송되는 파일의 손실없이 전송이 완료된다. 향후 본 설계를 기반으로 시스템이 구현이 필요하다.

REFERENCES

[1] K. I. Kim, J. S. Kim, “Big Data Processing and Performance Improvement for Ship Trajectory using MapReduce Technique”, *Journal of The Korea Society of Computer and Information*, Vol. 24 No. 10, pp. 65-70, Oct, 2019

[2] K. Shvachko, H. Kuang, S. Radia, R. Chansler, "The Hadoop Distributed File System. In Mass Storage Systems and Technologies (MSST)," *2010 IEEE 26th symposium on IEEE*, Vol.1, No.1, pp.1-10, 2010.

[3] B. H. Lee, D. M. Yang, “A Security Log Analysis System using Logstash based on Apache Elasticsearch”, *Journal of the Korea Institute of Information and Communication Engineering,* Vol. 22, No. 2, pp.382-389, Feb, 2018

[4] G. W. Jin, “A Study on the Data Collection Methods based Hadoop Distributed Environment”, *Journal of the Korea Convergence Society,* Vol. 7, No. 5, pp.1-6,

Oct, 2016
[5] K. S. Noh, S. T. Park. K. H. Park, "Convergence Study on Big Data Competency Reference Model", *Journal of Digital Convergence*, Vol. 13, No. 3, pp.55-63, 2015
[6] Y. H. Lee, J. H. Suh, "Big Data Platform for Utilizing and Analyzing Real-Time Sensing Information in Industrial Sites", *The Korean Society for Creative Information Culture*, Vol. 6, No. 1, pp.15-21, Apr, 2020
[7] V. Q. Nguyen, H. N. Nguyen, K. B. Kim, "Design of a Platform for Collecting and Analyzing Agricultural Big Data", *Journal of Digital Contents Society*, Vol. 18, No. 1, pp. 149-158, Feb. 2017
[8] J. M. Moon, K. S. Shin, "Measurement of Latency and Uplink Throughput According to Number of NB-IoT Devices", *The Journal of Korean Institute of Communications and Information Sciences* ,Vol.44 No.06, pp.1188-1192

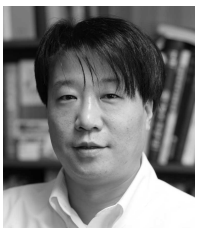
저자약력

최 종 석(Jongseok Choi) **[정회원]**

- 2013.03 ~ 2015. 02
  숭실대학교 박사수료
- 2019. 10 ~ 현재
  ㈜공감하다 대표
- 2020. 09 ~ 현재
  숭실대학교 스파르탄SW교육원 교수

〈관심분야〉 빅데이터, IoT, 데이터분석, 무선통신

신 용 태(Yongtae Shin) **[비회원]**



- 1994 Unviersity of Iowa Computer Science 공학박사
- 1995. 03 ~ 현재
  숭실대학교 컴퓨터학부 교수
- 2018. 03 ~ 현재
  숭실대학교 스파르탄SW교육원 원장

〈관심분야〉 인공지능, 데이터분석, 무선통신, IoT