ICIC Express Letters
Part B: Applications
Volume **13**, Number **8**, August **2022**

ICIC International ©2022 ISSN 2185-2766
pp. 837–845

A WORKFLOW-BASED ENGINEERING DATA ANALYTICS PLATFORM

Dongwoo Seo[1], Daeyong Jung[1,*], Myungil Kim[1] and Seok Chan Jeong[2,3]

[1]Korea Institute of Science and Technology Information (KISTI)
245 Daehak-ro, Yuseong-gu, Daejeon 34141, Korea
{ seodongwoo; mikim }@kisti.re.kr; *Corresponding author: daeyongjung@kisti.re.kr

[2]Department of e-Business
College of Commerce and Economics
[3]AI Grand ICT Research Center
Dong-Eui University
176 Eomgwang-ro, Busanjin-gu, Busan 47340, Korea
scjeong@deu.ac.kr

Received December 2021; accepted February 2022

Abstract. *There is an urgent need for a big data platform for storing, sharing, and analyzing mass-produced data for efficient data-intensive R&D. To solve this problem, this study developed a workflow-based big data platform which enables engineering experts without specialized knowledge in data analysis to solve engineering problems by using large-scale data collection and analysis technology. In this study, we design a workflow-based engineering data analytics platform. The objective of this study is to establish a workflow-based analytics system for the big data generated from engineering fields and develop a data analysis system using artificial intelligence for deducing meaningful results from the data collected in the big data platform. All these tasks are operated and managed based on the workflow. Analytical models modeled using components are analyzed through an analysis engine using machine learning. Our platform includes pre-processing, analysis, and visualization algorithms required for data analysis.*

Keywords: Machine learning algorithm, Workflow, Artificial intelligence, Engineering field, Big data platform

1. **Introduction.** Recently, innovation in the engineering industry using technologies based on the 4th industrial revolution, such as the Industrial Internet of Things (IIoT) and artificial intelligence, is an essential element in improving national manufacturing competitiveness. With the development of industrial Internet technology, real-time collection of various types of data that was previously impossible to obtain has become possible [1]. However, for big data analysis, it takes a lot of time and money to build an infrastructure to store and manage data, and it is difficult for many companies to utilize it due to the nature of the data analysis field, which is highly dependent on analysis experts [2]. In particular, in the field of engineering, it is difficult for data analysis experts to understand various domains related to engineering, and it is difficult for engineering experts to have expertise in data analysis, so there is a limit to the application of engineering using big data technology. In addition, there is an urgent need for a big data platform for storing, sharing, and analyzing mass-produced data for efficient data-intensive R&D.

To solve this problem, this study developed a workflow-based big data platform which enables engineering experts without specialized knowledge in data analysis to solve engineering problems by using large-scale data collection and analysis technology. This paper is organized as follows. Section 2 introduces related works on data analysis platform. Section 3 describes the design of the workflow-based engineering data analysis platform.

DOI: 10.24507/icicelb.13.08.837

Section 4 describes an application prototype of the proposed platform. Section 5 concludes this paper with expected effects and future research plan.

2. **Related Works.** Recently, as well as research on algorithms to solve manufacturing problems, research on data analysis platforms that collect data from the manufacturing process and perform continuous analysis using the data is also being conducted. From the Hadoop-based platform-related research for big data analysis, a system framework that can be used to solve manufacturing problems using the cloud computing concept and IoT (Internet of Things) was proposed [3,4]. In addition, for the engineering machine-learning automation platform, a decision-making system that can be implemented using machine learning methodologies on the cloud was proposed [5]. In addition, deep learning-based 1-D convolutional neural network is proposed using the time-sequence bearing data from the Case Western Reserve University (CWRU) bearing database [6].

In previous studies, research was conducted to solve manufacturing problems through various types of data analysis algorithms, and to use computing technology to perform the analysis process and build a system. However, many studies have limitations in that the process targeted for analysis is limited or it is difficult to collect, store, and analyze large-scale sensor data. Therefore, in order to solve the problems of the diverse and huge engineering industry, a platform that can easily collect, refine, and analyze large-scale data is required. To solve this problem, this study developed a workflow-based big data platform which enables engineering experts without specialized knowledge in data analysis to solve engineering problems by using large-scale data collection and analysis technology.

3. **Proposed Approach.** The workflow-based engineering data platform presented in this study signifies a High-Performance Computing (HPC) system that can store and manage the big data in engineering fields based on the cloud and can analyze and predict the data through an artificial intelligence library based on the workflow; it aims to discover a new business that fuses the big data and artificial intelligence technology. The schematic diagram of the target system is shown in Figure 1.

1) HPC based Cloud Computing: The system installed in the KISTI. It is composed of the interpretation, development, and design resource groups.
2) Bigdata Storage: It stores the collected raw data as big data.
3) Bigdata Management: It conducts the distributed processing of the data learning by linking with the big data and machine learning module.

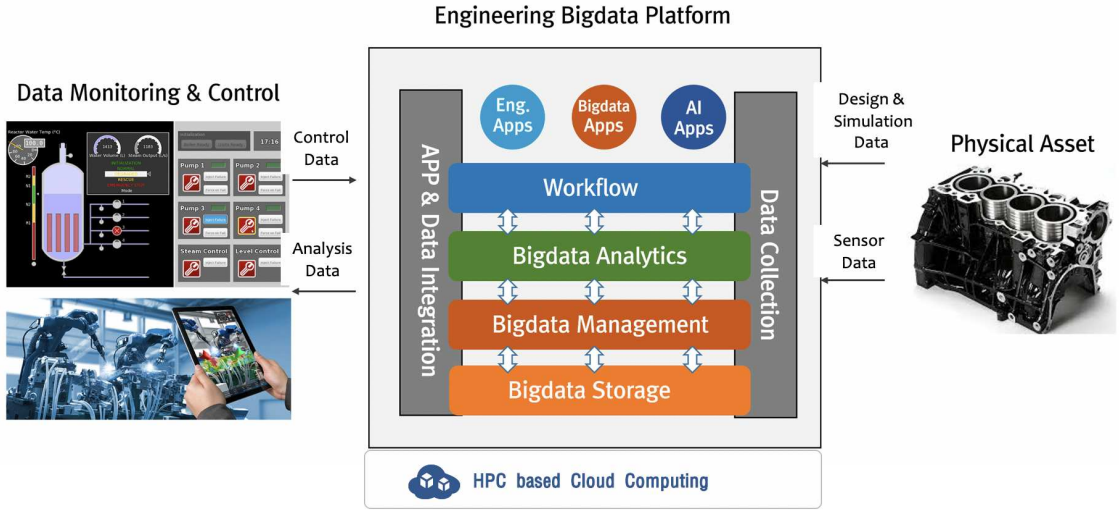

FIGURE 1. Proposed platform

4) Bigdata Analytics: It analyzes and learns the collected big data and generates the prediction data using machine learning.
5) Workflow: It is an analysis tool based on the open-source workflow and utilizes the KNIME [6]. It executes the workflow of the user at the big data platform and identifies the flow of the raw data by connecting with the monitoring system.

3.1. **Storage and management.** The big data platform consists of open-source software and is installed at the cluster composed in the cloud environment. The Cloudera [7] is composed of Hadoop, HDFS, Kafka, Zookeeper, Spark, Impala, and Hive. Figure 2 shows the composition of the open-source software components installed at the cluster composed of a total of five nodes.

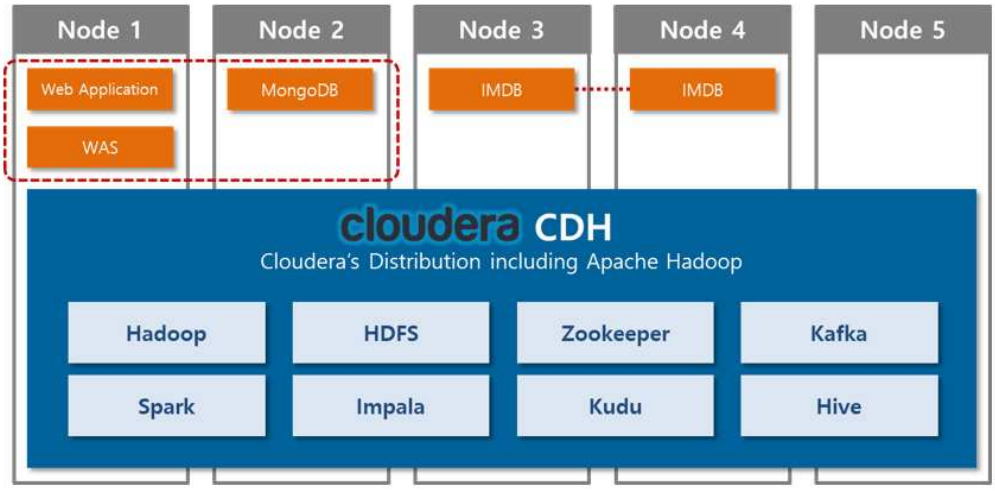

FIGURE 2. CDH architecture

The big data in which sensing and stream data are accumulated can be utilized in various fields. One of them is the artificial intelligence integrated system that can analyze and learn the previous data trend and predict the future data generation status. Therefore, the sensing and stream data should be continuously collected and the collected data should be stored in a structured form. For establishing such a system, the previously built big data storage/management platform is utilized in this study; particularly, Kafka and Impala are used for data collection and storage.

Impala was developed as a system that could analyze the data stored in the Hadoop platform in real time using SQL. It can obtain better-quality results using the self-distribution question engine without using the MapReduce framework of Hadoop. Although the existing HDFS-based relational database (e.g., Hive) could carry out the effective distributed processing of the massive data, many users who were not familiar with the distributed environment demanded the fast question and answer speed, similar to the database in the existing single-node environment. Impala improved the processing speed using the method of performing substantial distributed processing in the memory. Because it selected HiveQL, an SQL provided by Hive, as an interface and provided an interface that was not significantly different from the universal SQL, users could access the data in a familiar way.

3.2. **Workflow based Bigdata Analytics.** For the workflow-based big data analysis system, the open-source software KNIME, which is a workflow-based data analysis tool, is used. The data analysis work is generally carried out through the following procedures: 1) access to data, 2) data conversion, 3) data analysis, 4) visualization, and 5) distribution. KNIME provides all procedures in UI to intuitively identify these work procedures and easily develop new work.

Keras provided by KNIME as a method to perform deep learning is basically a library operating on Python. However, KNIME is a development tool developed by Eclipse-based Java; hence, the environment using Python should be established in advance to conduct the Keras function; the method of linking Python and the Keras node of KNIME should be analyzed to advance the Keras node.

To operate Python, KNIME generates the Python process using ProcessBuilder and obtains InputStream and OutputStream through the process handle. The InputStream and OutputStream of Python obtained are the standard input and output of the Python process; using InputStream and OutputStream, the script can be executed at an external application program through Python. The method of exchanging messages through the standard input and output at an external application program is referred to as Inter-Process Communication (IPC), and messages can be obtained through pipes.

After the data structure is generated at Keras Integration and transferred to the Python process, the delivered data structure is changed to Python script and executed at the Python process. Therefore, to modify the "Keras Network Learner" function, the UI and data structure of 'Keras Integration' should be revised and the script conversion of the 'Python Process' should be revised. The link structure of Keras-Python is shown in Figure 3.

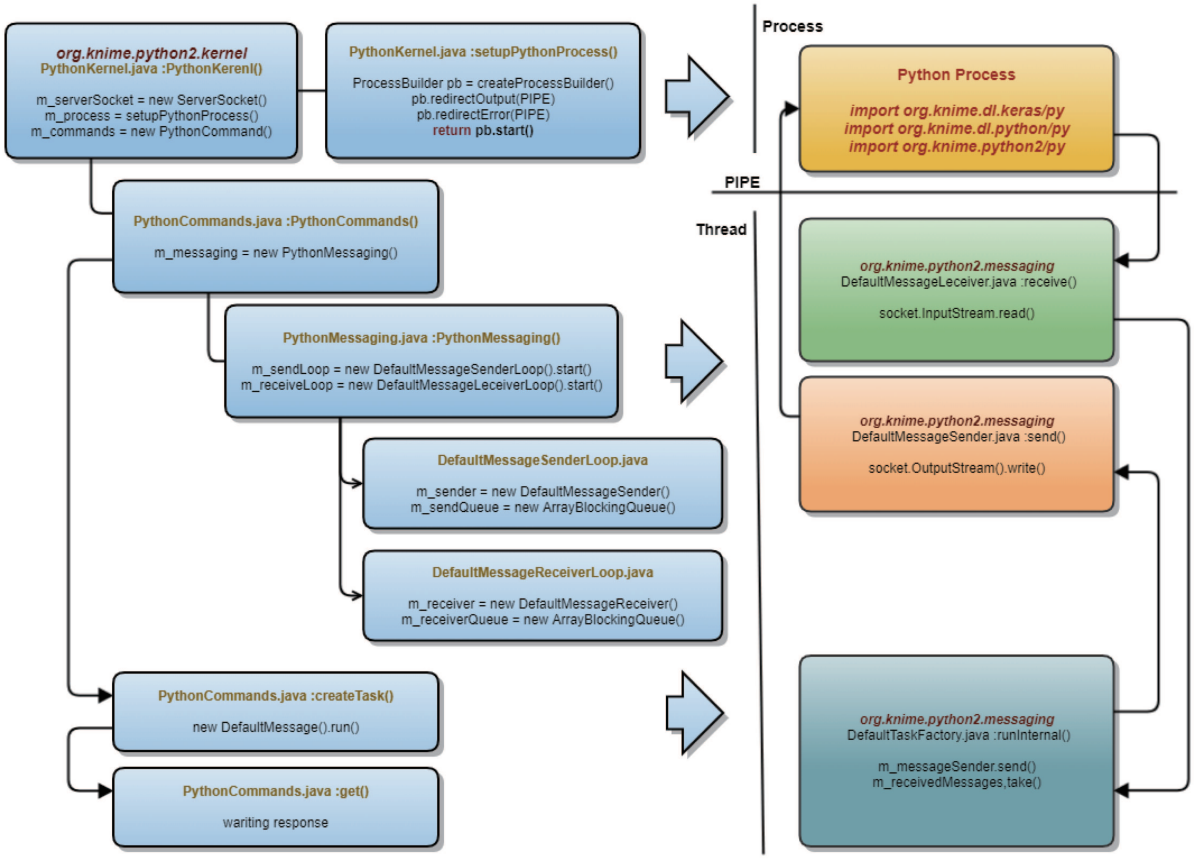

FIGURE 3. Keras-Python class diagram

Generally, deep learning first generates a model to learn data and then learns data using the loss function to determine the data weight alongside the metric function to determine the processing of data removed through the loss function. However, the metric function in Keras is not used for processing the lost data, but for inspecting the loss rate of data when learning using the entered loss or metric functions.

In KNIME, to support deep learning using Keras, nodes are separated into the learner for various layers, learning for modeling, and executor for prediction. The node for learning is referred to as "Keras Network Learner", and it sets up the loss function and input data for learning and conducts learning. The basic configuration screen of the "Keras Network Learner" node is shown in Figure 4.

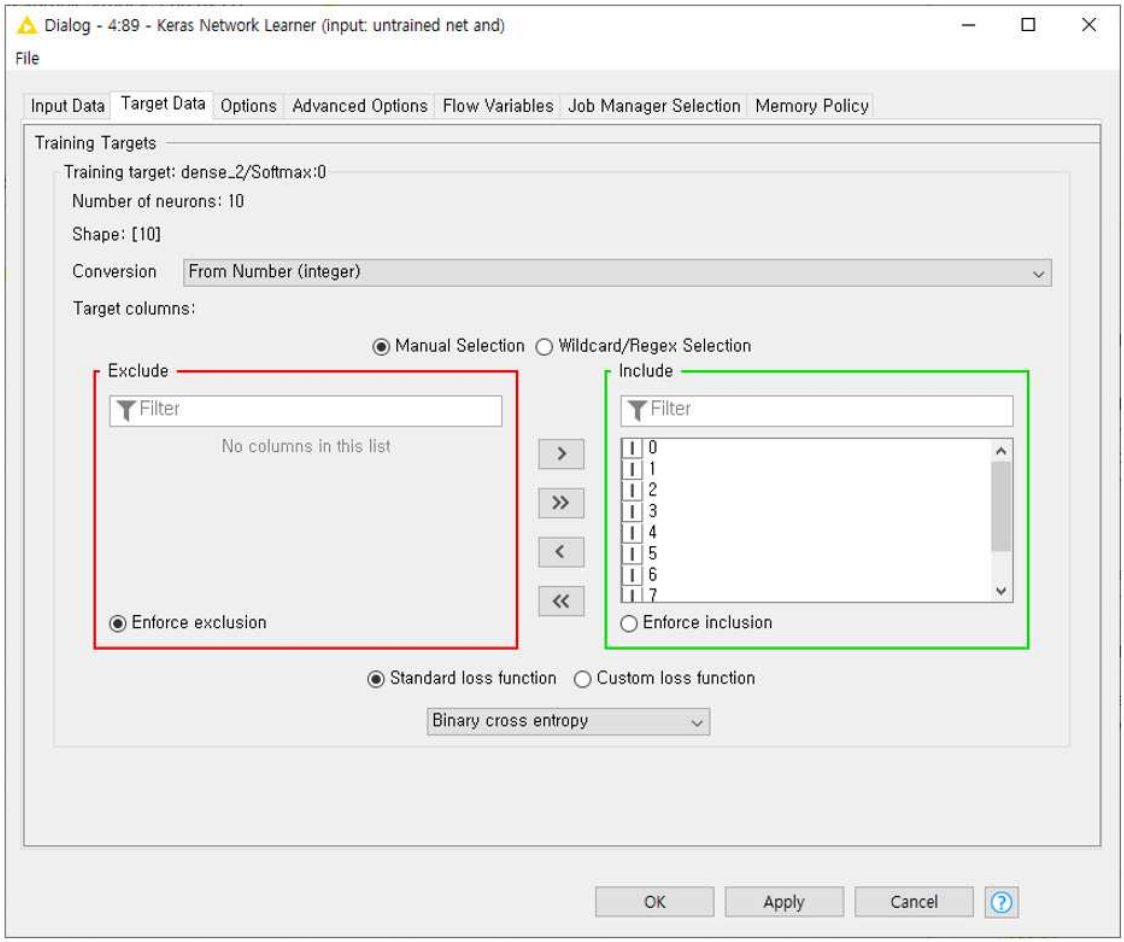

FIGURE 4. Keras Network Learner dialog (input data)

After selecting the input data and data type, the loss function can be set up through the drop-down menu of the "Standard loss function" at the bottom as shown in Figure 5.

The Metrics Function item was added to the Keras Network Learner node setting dialog as shown in Figure 6.

4. **An Application Prototype of the Proposed Platform.** The data used in this use case are the sensing data of the wave-power plant. A total of 172,000 data are obtained from 59 sensing data and their date-time information at 0.1 s intervals for 24 h a day. Although the Data Agent should transfer the data generated from the sensor to the Kafka Broker in real time, a system that transfers 172,000 data at 1 s intervals or specified intervals in real time is developed in this study.

This study aims to establish the system for the real-time monitoring of the entered sensing data, and the monitoring is performed after selecting specific data of the wave-power plant data. The column that conducts the monitoring is as follows.

First, the entity corresponding to the above column is generated and the properties required for this entity to recognize the data are designed. The protocol produced in the JSON form obtained after generating each entity and designating properties is shown in Table 1.

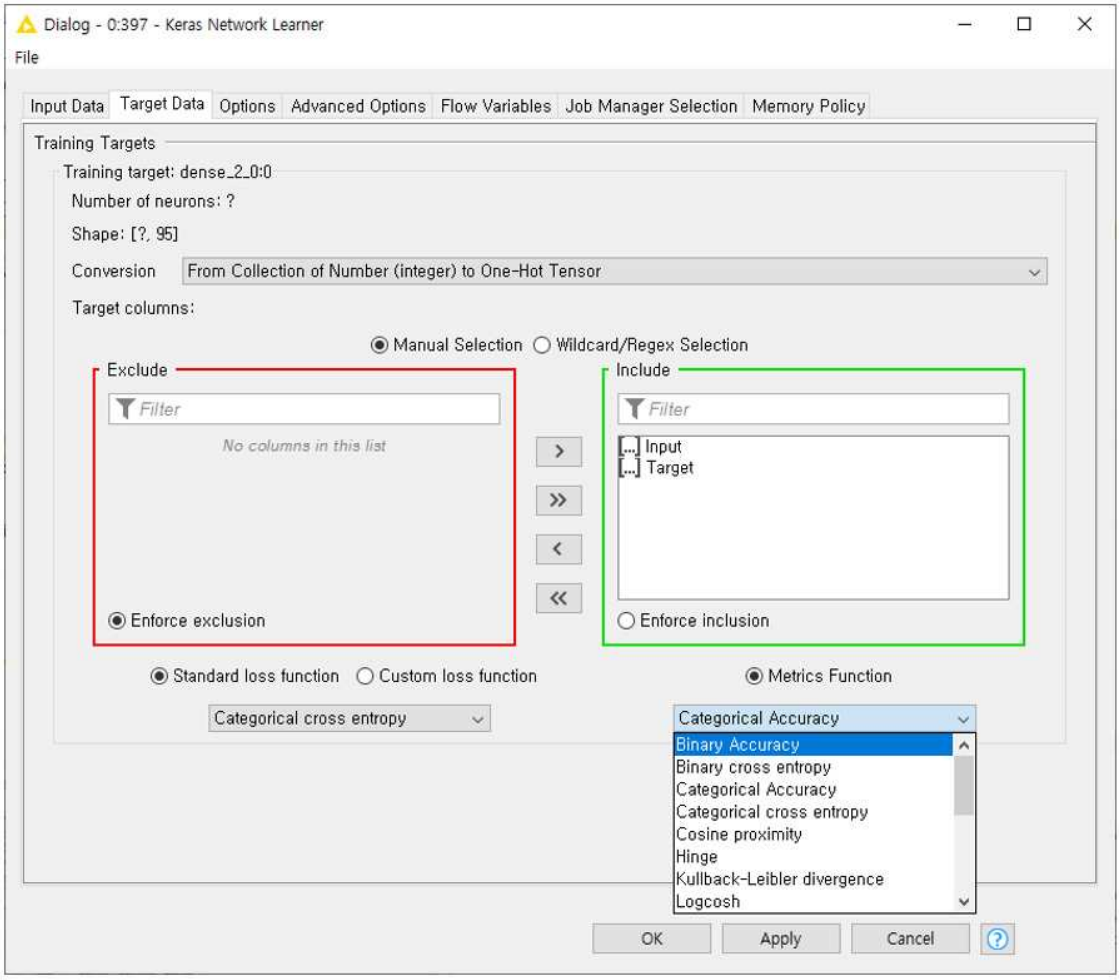

FIGURE 5. Keras Network Learner dialog (metrics function)

FIGURE 6. Deep learning progress

TABLE 1. Protocol example

Entity	Properties	JSON
Flow velocity	"WaveSpeed": present value "WaveSpeed_SG": present SG value	{"WaveSpeed": 11, "WaveSpeed_SG": 33}
Pressure L	"WavePress": present value "WavePress_PD": predicted value "WavePress_SG": present SG value "WavePress_SGPD": SG predicted value	{"WavePress": 11, "WavePress_PD": 22, "WavePress_SG": 33, "WavePress_SGPD": 44}
Pressure R		
Torque	"WaveTorque": present value "WaveTorque_SG": present SG value	{"WaveTorque": 11, "WaveTorque_SG": 33}
RPM	"WaveRPM": present value "WaveRPM_SG": present SG value	{"WaveRPM": 11, "WaveRPM_SG": 33}
Power	"WavePower": present value "WavePower_SG": present SG value	{"WavePower": 11, "WavePower_SG": 33}

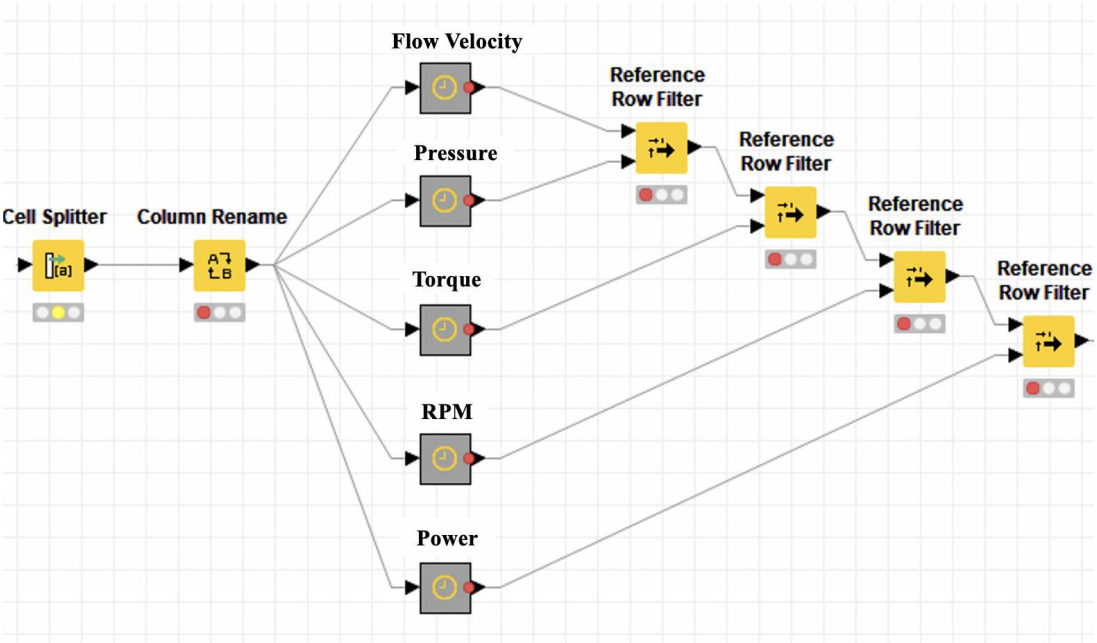

FIGURE 7. Workflow for analysis data

The Kafka Consumer connects to the Kafka Broker and receives the message of the desired topic. During this process, it does not receive one message but all data after the previously received data; thus, it successively analyzes the received data as shown in Figure 7.

Because the data received at the Kafka Broker is composed of one string data, they should be separated into character strings in meaningful units. In this service, the input data are in CSV form; hence, the character strings are distinguished using ',' as a separator through the procedure shown above. The separated character strings are converted to the table and column forms to be processed at KNIME as shown in Figure 8.

Among the character strings separated by ',', the column that should be extracted for monitoring, namely the 'flow velocity 1-6' column, has a column name in Korean; thus, the column name should change to an English name. Here, the column name is changed to the key fit to the properties of the flow velocity. The work procedure is carried out in parallel to simultaneously process all data from flow velocity 1 to flow velocity 6.

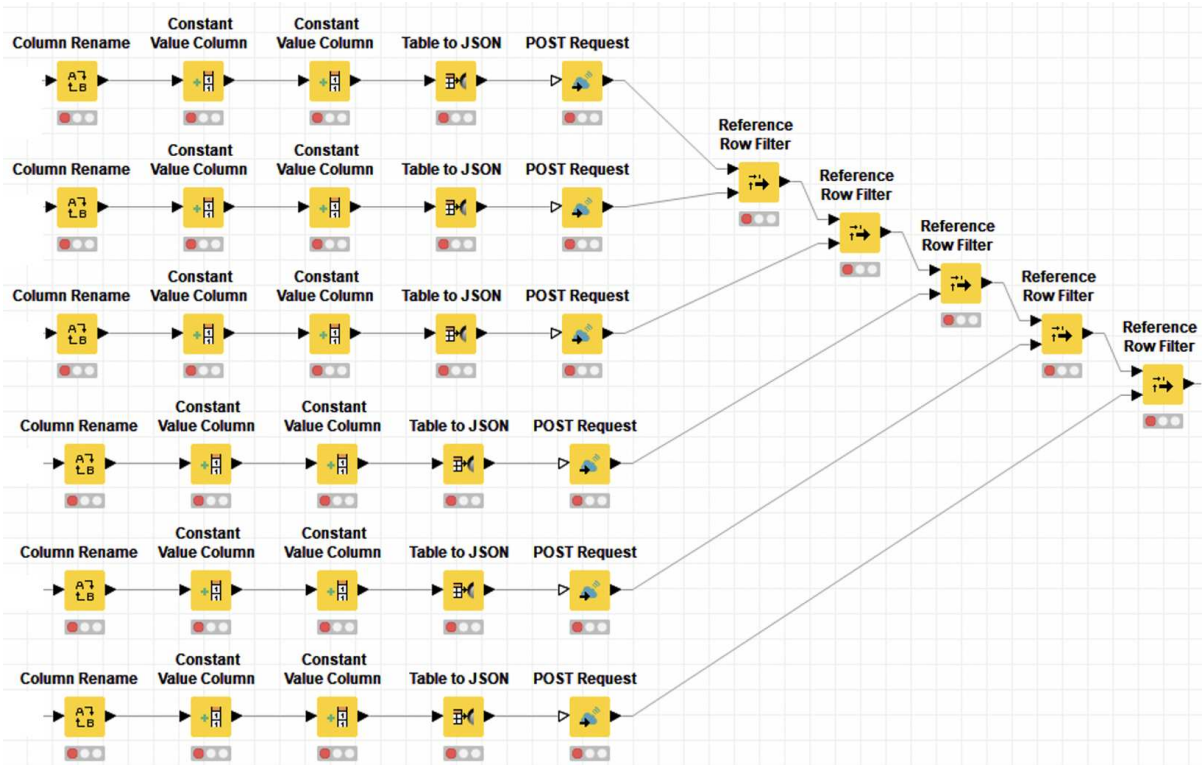

FIGURE 8. Workflow for conversion data

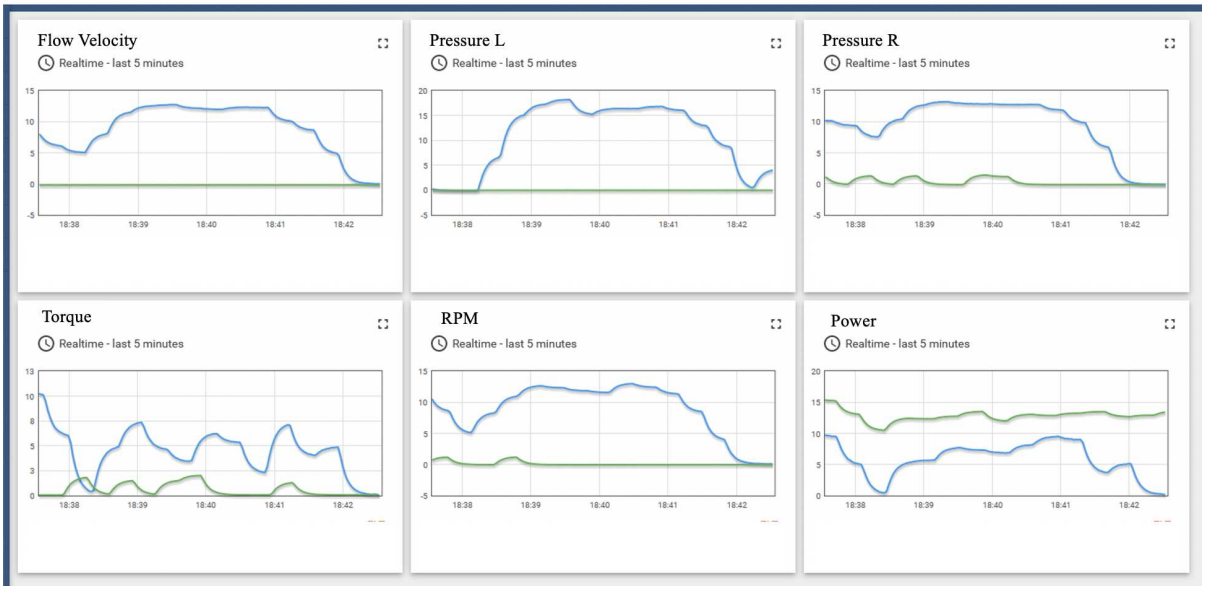

FIGURE 9. Data visualization

The columns corresponding to the device of the corresponding dashboard are extracted at the table and converted to JSON. Next, the REST API of dashboard is called by calling the POST Request node. The setup-completed dashboard and monitoring screen results are shown in Figure 9.

5. **Conclusions.** This study proposed a big data platform using workflow that easily stores, manages, and analyzes large-capacity big data generated in the engineering field. To this end, we designed a system using high-performance computing in the cloud environment and proposed a platform that can be used industrially by engineering experts who do not have expertise in data analysis to collect and analyze IoT big data through the

engineering big data platform. For future work, we are improving the platform making it capable of analyzing a variety of engineering data, and to be able to use an HPC cloud platform to share workflow among users.

Acknowledgment. This research was supported by Korea Institute of Science and Technology Information (KISTI) and this research was a part of the project titled 'Development of a ICT based safety technology for Trawl (20210494)', funded by the Ministry of Oceans and Fisheries, Korea.

REFERENCES

[1] D. Batanov, N. Nagarur and P. Nitikhunkasem, EXPERT-MM: A knowledge-based system for maintenance management, *Artificial Intelligence in Engineering*, vol.8, no.4, pp.283-291, 1993.

[2] Z. Bi, L. D. Xu and C. Wang, Internet of Things for enterprise systems of modern manufacturing, *IEEE Trans. Industrial Informatics*, vol.10, no.2, pp.1537-1546, 1996.

[3] A. Djuraidah, R. N. Rachmawati, A. H. Wigena and I W. Mangku, Extreme data analysis using spatio-temporal bayes regression with INLA in statistical downscaling model, *International Journal of Innovative Computing, Information and Control*, vol.17, no.1, pp.259-273, DOI: 10.24507/ijicic.17.01.259, 2021.

[4] C. Michele, Smart manufacturing technology, *Applied Sciences*, vol.11, no.12, DOI: 10.3390/app11178202, 2021.

[5] S.-W. Choi, E.-B. Lee and J.-H. Kim, The engineering machine-learning automation platform (EMAP): A big-data-driven AI tool for contractors' sustainable management solutions for plant projects, *Sustainability*, vol.13, no.18, DOI: 10.3390/su131810384, 2021.

[6] D. Neupane, Y. Kim, J. Seok and J. Hong, CNN-based fault detection for smart manufacturing, *Appl. Sci.*, vol.11, DOI: 10.3390/app112411732, 2021.

[7] N. Karimah and G. Schaftenaar, KNIME-based analysis of off-target effect of drugs related to the molecular 2D fingerprint, *Pharmaceutical Sciences*, vol.24, pp.137-266, DOI: 10.18433/jpps31771, 2021.

[8] A. Hong, W. Xiao and J. Ge, Big data analysis system based on Cloudera distribution Hadoop, *2021 7th IEEE Intl Conference on High Performance and Smart Computing (HPSC)*, DOI: 10.1109/BigDataSecurityHPSCIDS52275.2021.00040, 2021.