Artificial Intelligence Platform Based for Smart Factory

Aji Teguh Prihatno
Electrical Electronics Engineering
Kookmin University
Seoul, South Korea
aji.teguh@gmail.com

Himawan Nurcahyanto
Electrical Electronics Engineering
Kookmin University
Seoul, South Korea
himawanurcahyanto@gmail.com

Yeong Min Jang
Electrical Electronics Engineering
Kookmin University
Seoul, South Korea
yjang@kookmin.ac.kr

***Abstract*— The effect of Artificial Intelligence (AI) on smart manufacturing is increasing rapidly. The AI-based open-source web application enables users to build and distribute live code, calculations, visualizations, and explanatory text documents, and also supports the data cleaning and conversion, numerical simulation, mathematical modelling, machine learning, and much more are used by the industries. This paper describes the processing data particle $PM_{2.5}$ and creates prediction using an AI algorithm from those data which the result can be implemented in the smart factory. To support the accuracy of the prediction of the AI method in the smart factory environment, the author uses Jupyter Notebook based on the source web application. This deployment will lead to performance improvement, cost reduction, process management, shortened product cycle production times, and increased productivity for the manufacturing sector.**

Keywords*— *Artificial Intelligence (AI), AI Platform, $PM_{2.5}$, Smart Factory, Manufacturing, Jupyter Notebook

I. Introduction

The advancement of Artificial Intelligence (AI) technology has continued to grow rapidly especially in the field of a smart factory. AI is one of the main key roles in smart factories that will lead the market size to expand. From statistics [1]. This shows us the increase in the revenue of the AI global market from 2018 to 2024 which is estimated to be around 126 billion USD as shown in Figure 1 below.

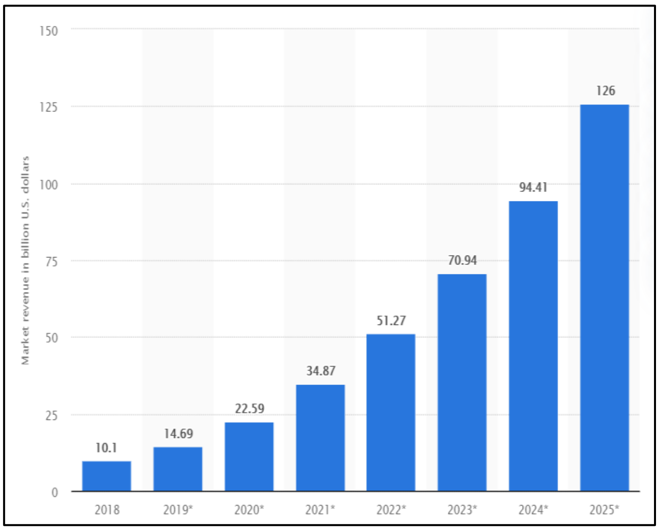

Figure 1. Revenues from the AI software market worldwide from 2018 to 2025 *(in billion U.S dollars)*

The increment of AI global market as Figure 1 shows, driven by three factors. The first factor, the growth of demand for the application of predictive maintenance and machinery inspection, which are widely spread usage of computer vision cameras in machinery inspection. The second factor, the implementation of the Industrial Internet of Things (IIoT), and the third factor is the use of big data in the manufacturing industry. This growing of AI global market needs to reduce operating costs and machine downtime also complements the growth of the application of predictive maintenance and machinery inspection in industries [2].

As a big element of this digital transformation, AI is being touted. Even with new innovative Manufacturing 4.0 innovations, the majority of connected devices in manufacturing, including initiation, management, tracking, and feedback, are still unable to make decisions without human intervention. Infusing knowledge into these physically linked things will increase the value that can be produced from them exponentially. The purpose of a smart factory is endorsed by AI; one that will function with minimal human contact [3].

In addition to improving the environment and the quality of indoor air quality in smart factories, AI-based smart factory can also optimize efficiency, quality, cost, and resource management processes at the global production level. It is also directly proportional to the size of the AI market that drives the growth of the size of the smart factory market.

This paper explains a smart factory that runs the AI Platform, related to the measurement of environmental values in the plant, such as Particulate Matter ($PM_{2.5}$).

II. AI Platform Based on Open Source

To process and train the data of $PM_{2.5}$ in the Smart Factory, the authors have implemented the AI algorithm using Jupyter Notebook. The Jupyter notebook is based on a set of collaborative computing open standards. For interactive computing on the internet think HTML and CSS. Third-party developers will utilize these open standards to create customized applications with embedded interactive computing.

With its modular framework, Jupyter Notebook expands the notebook to visualization, multimedia, collaboration, and more beyond code. It stores code and output, along with markdown notes, in an editable document called a notebook, in addition to running the code. This is sent from browser's users to their notebook server when they save it which stores it as a JSON file with a.ipynb extension on the disk. The design architecture of Jupyter Notebook and its interface can be seen in Figure 2.

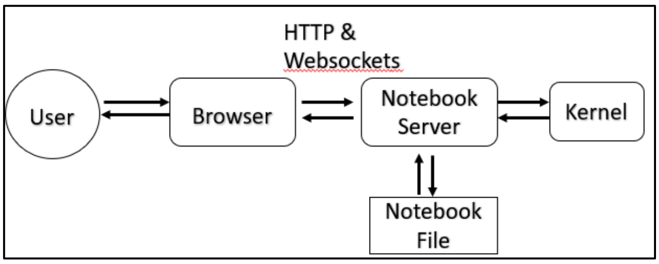

Figure 2. The Jupyter Notebook Interface

The notebook server is responsible for saving and loading notebooks, so even if the user doesn't have the kernel for that language, the user still can edit notebooks. There's nothing the kernel knows about a notebook document: it simply gets code cells sent to run when the user runs them [4]. The kernel of a notebook is a "computer engine" that executes the code found in the Notebook file. A python code is executed by the ipython kernel which kernel exists for many other languages (official kernels). The associated kernel is automatically launched when the user opens a Notebook File. The kernel performs the computation and generates the results when the notebook is executed (either cell-by-cell or with the menu Cell -> Run All). The kernel can consume important CPU and RAM depending on the type of computation. Remember that the RAM is not released until the user has shut down the kernel.

III. IMPLEMENTATION AND RESULT

To predict $PM_{2.5}$ concentration in a factory environment, the authors were using RNN (Recurrent Neural Network) that we can shape several sets of sequences, from the data using time from a continuous data set, and also from certain sets of data sequences, we can observe the correlation between sequences. To form a network, Simple RNN has several neuron-nodes. Per node (neuron) has a real-valued activation that varies in time. Each relation has a real-value weight, which can be modified in each case. [5].

The human presence in the cleanroom is the most easily observed associations in $PM_{2.5}$ data, according to the aim of this research is to get the smart factory environment. The $PM_{2.5}$ concentration will also increase as more individuals come into the cleanroom.

After processing the data using Jupyter Notebook, we get the experimental result to predict one day ahead of the concentrate of $PM_{2.5}$, in respectively, the accuracy of Mean Absolute Error (MAE) 0.08, Root Mean Square Error (RMSE) 0.11, and Mean Square Error (MSE) 0.01.

As we can see from Figure 3, the graph of prediction in the blue line is near to the real data which the line is red.

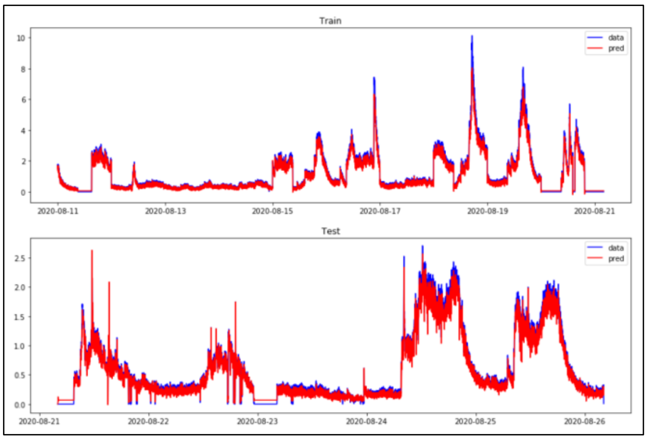

Figure 3. Result Implementation. Prediction $PM_{2.5}$ using RNN Method.

IV. CONCLUSION

The purpose of the work presented was to implement AI by building prediction machines as the basis for the smart factory with time-series data. The RNN model reveals substantial results in the long-term $PM_{2.5}$ concentration based on historical data. Nevertheless, to improve the accuracy of the prediction machine and extend the reach of the smart factory operated by the AI network, the model needs to be further and more varied in the future. Finally, the prediction of $PM_{2.5}$ status will assist operators in the operational policy and allocation of smart factory capital.

ACKNOWLEDGEMENT

This work was supported by the Ministry of Trade, Industry, and Energy (MOTIE) and Korea Institute for Advancement of Technology (KIAT) through the International Cooperative R&D program (Project ID:P0011880)

REFERENCES

[1] Statista, "'Global size of the smart factory market in 2019 and 2024,'" *Online*. https://www.statista.com/statistics/872289/worldwide-smart-factory-market-size/.

[2] M. and Markets, "Artifical Intelligence in Manufacturing Market," 2020. https://www.marketsandmarkets.com/Market-Reports/artificial-intelligence-manufacturing-market-72679105.html.

[3] J. Lee, J. Singh, and M. Azamfar, "Industrial artificial intelligence," *arXiv*, 2019.

[4] Jupyter, "Jupyter Notebook Architecture." https://jupyter.readthedocs.io/en/latest/projects/architecture/content-architecture.html.

[5] Y. T. Tsai, Y. R. Zeng, and Y. S. Chang, "Air pollution forecasting using rnn with lstm," *Proc. - IEEE 16th Int. Conf. Dependable, Auton. Secur. Comput. IEEE 16th Int. Conf. Pervasive Intell. Comput. IEEE 4th Int. Conf. Big Data Intell. Comput. IEEE 3rd Cyber Sci. Technol. Congr. DASC-PICom-DataCom-CyberSciTec 2018*, pp. 1068–1073, 2018, doi:10.1109/DASC/PiCom/DataCom/CyberSciTec.2018.00178.