Journal of Information Technology Management

Print ISSN: 2008-5893

Online ISSN: 2423-5059

Integrated Model-Based Engineering using Deep Learning with IIoT for Industry 4.0

P. Senthilkumar *

*Corresponding Author, Department of Instrumentation and Control Engineering, Kalasalingam Academy of Research and Education, Tamilnadu, India. E-mail: senthilpsk.mail@gmail.com

K. Rajesh

Department of Electrical and Electronics Engineering, Kalasalingam Academy of Research and Education, Tamilnadu, India. E-mail: k.rajesh@klu.ac.in

Abstract

The Industrial Internet of Things (IIoT) is a potential platform for developing industry 4.0 and its related applications, especially in cyber-physical systems. Such a new trend in manufacturing sectors offers further potential to optimize operations, realize business models, and reduce costs. Such accomplish may also lead to complex and complicated tasks; hence, to deal with such issues, Reference Architecture Model Industry 4.0 (RAMI 4.0) is developed to structure Industry 4.0. In this paper, the standardized framework is considered RAMI 4.0 and its integration with an IIoT software named Software Platform Embedded Systems (SPES). Integrating Model-Based Engineering (MBE) with a framework requires using a deep learning model called Recurrent Neural Network (RNN). The RNN-MBE, which optimizes the entire process, is responsible for optimizing the process and reducing industry costs. The optimization problem has been fixed, and the MBE simulation has shown that using the proposed MBE is efficient.

Keywords: Model-based engineering; Recurrent neural network; Industrial Internet of Things (IIoT); Deep learning.

Journal of Information Technology Management, 2023, Vol. 15, Special Issue, pp. 112-123
Published by University of Tehran, Faculty of Management
doi: https://doi.org/ 10.22059/jitm.2023.91571
Article Type: Research Paper

Received: December 22, 2022
Received in revised form: January 15, 2023
Accepted: February 20, 2023
Published online: March 15, 2023

Introduction

Internet of Things (IoT) refers to a network architecture in which commonplace objects are connected to the Internet through wireless networks. The IoT can achieve its objectives by putting thousands of intelligent devices in place. These devices take in information from their surroundings, process it according to the requirements, and then transmit it using reliable and secure communication channels. Recent advancements in hardware, software, and communication systems have significantly contributed to improving human lifestyles (Singh et al., 2022; Zhang et al., 2021), particularly in terms of the amount of time, energy, and money that may be saved.

If an IoT concept is implemented in a business or manufacturing environment, it is referred to as the IIoT. The IIoT improves production processes by enabling technology in the workplace that is both environmentally benign and cost-effective (Mantravadi et al., 2022; Suthar & He, 2021). There has been tremendous growth in the IoT sector, and numerous sectors are adjusting to the digital transformations that are occurring by utilizing the technology available through IIoT. Strong congruences and alliances of interests between IIoT stakeholders and those developing applications have encouraged companies worldwide to invest in IIoT.

During this period of rapid technological advancement, IIoT systems' interconnectivity (the fourth industrial revolution) has exploded. The capability to intelligently process vast volumes of data is critically important for developing intelligent applications for the IIoT. Therefore, cognitive information processing frameworks are required to analyze massive amounts of data in IoT networks. Incorporating Artificial Intelligence (AI), and more specifically, Deep Learning (DL) techniques, into an IIoT system may prove to be extremely helpful (Brecher et al., 2021; Khalil et al., 2021).

DL techniques, for instance, make it possible for the system to learn from the lessons it has already been taught. The characteristics and nature of an algorithm dataset and its capacity to self-train are all essential factors in determining how well it can learn (Binder et al., 2021). Choosing the correct algorithm for deep learning might be difficult because of the possibility of algorithm selection challenges. Consequently, it is of the utmost importance to have a solid understanding of the DL workflow, its implementation, and many other possibilities. The significance of DL can be better understood by examining recent, ground-breaking research in the DL applied to IoT and IIoT applications.

Researchers carried out various research perspectives on DL-based and Industrial Internet of Things applications. However, the material that is now available has several significant shortcomings (Almadani & Mostafa, 2021). To begin with, most of these studies concentrated more on the theoretical aspects of DL for IIoT than the real-time application. Another criterion for an IIoT reference design is that it contains critical supporting technologies like cloud computing and big data analytics. It is necessary because these technologies are essential to the functioning of the IIoT. They also offered a minimal explanation of the use cases of the DL algorithms for the applications of the IIoT. Because of this, a comprehensive analysis of DL technologies for use in IIoT applications is provided to circumvent the many different constraints. This paper considers the standardized framework named RAMI 4.0 and its integration with an IIoT software called Software Platform Embedded Systems (SPES). Integrating Model-Based Engineering (MBE) with a framework requires using a deep learning model named Recurrent Neural Network (RNN)

Literature Review

In recent research, applications for IoT and IIoT have been proposed by people in academia and industry who are adopting DL-based solutions. Some recent survey articles will be discussed here to illustrate the depth and scope of the research that has already been done. A comprehensive discussion of DL techniques and their usefulness for the study of vast amounts of data, with a particular emphasis on IoT applications. In addition, several of the most well-known DL approaches and their beginnings and overall structures are dissected in great detail. DL techniques were also summarized in fog and cloud-based Internet of Things (Balachander et al., 2021).

In the meantime, Ma et al., 2019, carried out an in-depth study to determine whether or not it would be possible to implement DL strategies in IoT applications. This study's primary objective is to demonstrate how approaches from deep learning may be applied across a range of business sectors, including intelligent healthcare and manufacturing.

Deep learning (DL) has been demonstrated to help solve many issues, such as perspective and prediction analysis, power systems, financial time-series forecasting, and medical image processing (Boulila et al., 2021). Ambika, 2020, offered an in-depth assessment of ML and DL techniques, including their topologies as well as the influence these approaches have had on IIoT.

Implementing deep learning (DL) strategies in IoT applications might prove beneficial. The IoT has inspired the creation of a variety of real-time applications. These apps make use of IoT sensors to collect data about individuals and the environments in which they live. This information is input into DL models, improving an application's intelligence and capability level (Saleem & Chishti, 2021). Various deep learning algorithms are studied for real-time IoT applications. The writers of the work provided a wide range of DL tools and frameworks so that readers would have an easier time understanding DL models (Binder et al., 2020).

Methodology

In this paper, the main aim is to find a solution to the problem in industry 4.0 that uses IIoT as an input data acquisition tool. The block diagram of the proposed model is depicted in Figure 1. In this regard, the proposed method is modelled as two different modules: the sensing and control planes. The sensing plane encompasses IoT devices, and the control plane contains the central RNN controller for task allocation, where RAMI 4.0 is integrated with SPES. The task allocated to the systems in the industry is assigned based on availability. It is carried out by the presence of RNN in support of the RAMI 4.0, where the SPES-MBE platform acts as a supporting platform to process the input data from the IIoTs.

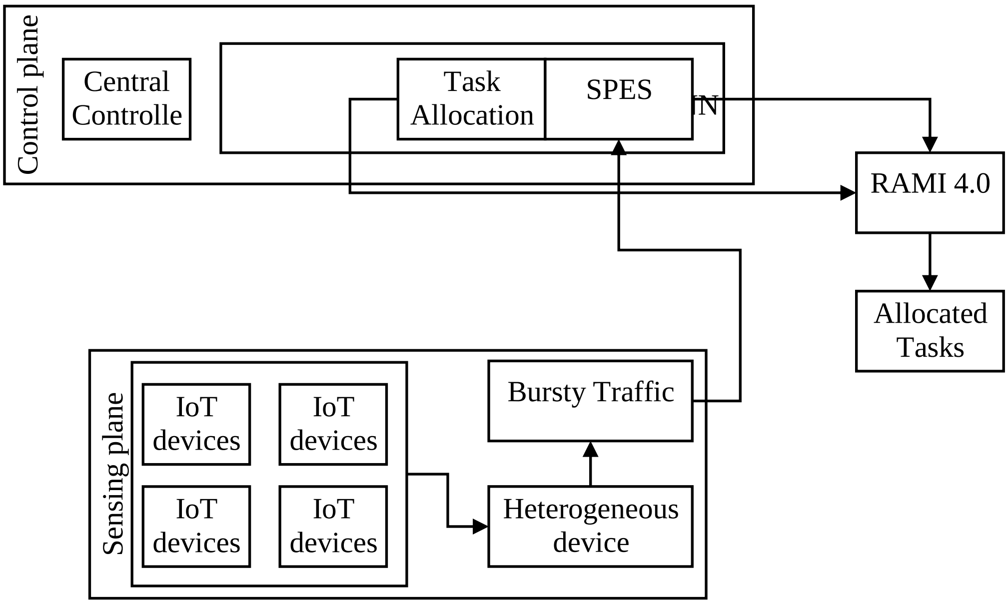

Figure 1. Block diagram of the proposed Model

Model-Based Engineering

This research work combines the ideas of RAMI 4.0 and SPES to produce an architecture that is more diverse and detailed for both existing industrial systems and those that will be developed in the future. Because both strategies employ a multi-layered structure, a connection between the two must be forged. On the other hand, it is essential to locate a solution that can fill in the voids that have been found. It would be beneficial to combine these two frameworks so that they can draw on the capabilities that each possesses when it comes to the creation of manufacturing at various levels of abstraction. As a result, the two frameworks are combined and mapped out by ISO 42010. Next, the top-down MDA development method is utilized to design an umbrella architecture that considers both frameworks' perspectives.

The architectural frameworks used in the various domains were initially mapped to the SPES framework. Nevertheless, RAMI 4.0 receives particular focus in this section because the approach in question is designed with the business world in mind. Because the labels of these two ideas suggest a straightforward one-to-one mapping between them, mapping the Business Layer to the Requirements Viewpoint is made to be relatively short. You can follow the same mapping technique when mapping the function layer with SPES' Function Viewpoint. On the other hand, when considering RAMI 4.0 Information and Communication Layer, the SPES matrix does not provide a perspective.

Because of this, it is essential to develop a better mapping technique to interface the two modelling frameworks. The two RAMI 4.0 layers are mapped to one another using the Technical Viewpoint available in SPES. The fact that the interfaces and protocols for data transmission between components have a technical nature raises the question of why this is the case in the first place. Because it would be more challenging to acquire all of the pertinent information from the Technical Viewpoint, the mapping has been designed to only go in one direction. As a result of the Asset Layer being incorporated into this SPES viewpoint, the previously described process is complicated because additional information has been appended to the technical representation. When the Integration Layer inherits all of the AAS information, it becomes the Digital Twin representation with access to data from several perspectives. It is because the integration layer has all of the AAS information. This layer encompasses all RAMI 4.0 viewpoints, including the Functional Viewpoint, the Logical Viewpoint, and the SPES Technical Viewpoint.

Architecture

When describing a system architecture with ISO 42010, the first step is to provide a vantage point for the numerous concerns important to the various stakeholders. Therefore, the first thing that needs to be done is to identify the stakeholders and their problems with the functional architectures. The following examples are intended to serve as a concise summary of the results of the process. However, function developers and process engineers care more about a comprehensive functional description that includes inputs and outputs and their interrelationships. Requirements engineers are concerned with precisely formalizing them, but function developers and process engineers care more about a complete functional description. In addition, the manager's primary focus is on satisfying the customers' requirements, whereas the network administrator must comprehensively explain every technological component. The SPES framework has now defined the views; the next stage, which will allow the architectural theory to be realized, is to declare the model types for each viewpoint.

Algorithm

Step 1: Interconnect RAMI with SPES, i.e. cross-domain modelling is done by mapping a domain framework (industry 4.0) with SPES.

Step 2: Map business layers with viewpoints (requirement) for one-one transformation

Step 3: Map functional layer with viewpoints (available) for one-one transformation

Step 4: Disable communication and information layer

Step 5: Map RAMI 4.0 layers with Technical Viewpoint

Step 6: Exchange data via protocols or components

Step 7: Map Asset layer with SPES viewpoint

Step 8: Add additional technical representation during data communication

Check if the scheduling is of data collected from IIoT to the target environment via RNN

Integration

Incorporating SPES's one-of-a-kind characteristics into RAMI Toolbox is the culmination of the process that combines the two systems. The traditional SPES matrix layout needs to be accessible to users as soon as possible. A second user interface with a step-through format, which can be seen in Figure 2, was developed to achieve this goal.

Each row and column represent a different perspective and level of abstraction associated with the SPES. These colours represent the RAMI layers. In the Requirements Viewpoint, there is only a small selection of models available to address the concerns raised by the stakeholders. After clicking on any of the squares to ensure this occurs, the Add Model feature becomes visible. Both an allocate function and a decompose function are added in the same development phase. Consequently, it is now feasible to carry out supervised model modifications, as described by MDA. It launches a new tab in the browser where you may enter the path that connects the pieces in the Model.

Deep Learning Model

A directed graph that moves in chronological order is formed by the nodes that make up the graph of an RNN. Because of this property, it can exhibit dynamic behaviour throughout the period. The inputs and outputs are entirely disconnected in a typical neural network, often known as a CNN. In certain circumstances, past facts are required to predict the following word or statement accurately. As a consequence of this, it is essential to maintain a record of knowledge from the past. The RNN inclusion of a hidden layer can remedy this issue. A necessary characteristic of an RNN is its hidden state, which remembers the data's order.

Throughout the operation, RNN is equipped with a memory that logs and catalogues all the information it has amassed. The parameters and information regarding the calculations performed are the same for all inputs and outputs. It uses the same settings and completes tasks on all inputs and hidden layers to produce a suitable outcome. It ensures that the work will be accurate. This characteristic makes the parameters more straightforward to understand than those of other neural networks.

RNNs are the most popular form of artificial neural network, but there are many other kinds. It can exhibit dynamic behavior throughout the period. In a typical neural network, there is no connection between any two of the network inputs or outputs. In certain circumstances, past facts are required to predict the following word or statement accurately. As a consequence of this, it is essential to maintain a record of knowledge from the past. The RNN inclusion of a hidden layer can remedy this issue.

The capability of an RNN to recall past sequences performed by a user is among its most essential characteristics. The RNN is equipped with its memory, which saves all of the knowledge acquired due to the calculations. It uses the same settings and performs tasks on all inputs and hidden layers to produce a suitable output. It ensures that the work will be accurate. This characteristic makes the parameters more accessible to understand than those of other neural networks.

This section demonstrates the functionality of RNNs by way of a straightforward example. The study shows that the RNN consists of one input, four hidden, and an output layer. The layers consist of weights and biases, and the analysis assumes the preferences (b1, b2, b3) and consequences over the hidden layer are w1, w2, and w3. Because none of them can recall what came before, they cannot cooperate.

When this occurs, it is possible to include the weights and biases of all three hidden layers into a single recurrent layer. This section contains a formula that can be used to determine the current condition, as shown in Equations (1) and (2).

$$\mathrm{h_t = f(h_{t-1}, x_t)} \quad (1)$$

Here

ht - current state, ht-1 - previous state and xt - input state.

With such an expression, a hyperbolic tangent activation function can now be utilized.

$$\mathrm{h_t = tanh(w_{hh} h_{t-1}, w_{xh} x_t)} \quad (2)$$

Where, whh - current neuron weight and wxh - input neuron weight.

The output can then be calculated after this expression has been applied.

$$\mathrm{y_t = w_{hy} h_t} \quad (3)$$

Where, yt - output of each layer and why - output layer weight.

Results and Discussion

During the validations, an RNN model is used for scheduling, and the modifications made to the implementation can be checked based on the acquired findings. A simulation is being run to determine whether or not there are any advantages to integrating MBE with IIoT. This method is evaluated and compared to the benchmarking strategy when testing the proposed solution for deeper learning.

The monitoring of the input signal from the IIoT is looked to determine whether or not it occurs with decreased pumping activity timing and voltage. In addition to the research that focuses on cloud resources and IIoT source data collection, additional research investigates the function of cloud-IoT modelling.

Figure 2 illustrates the computational time required for scheduling a job using RNN that helps optimize the entire task in the standardized RAMI 4.0 framework. An analysis of Figures 2a and 2b demonstrates that the optimum scheduling of jobs in the whole platform has improved the rate of task allocation, which has resulted in a reduction in the amount of time required for calculation.

As can be seen in Figures 3a and 3b, it is seen that the RNN-MBE offers reduced cost for a job ($) while scheduling the appointment based on the inputs from the IIoT. The simulation results show that the proposed method achieves a reduced cost rate to schedule a task than the existing methods in the case of IoT systems. It is mainly selected to allocate the job in a standardized RAMI 4.0 framework by a SPES that processes the IIoT data acquisition. Compared to the existing methods, the solution-optimized array results in significantly lower operating expenses.

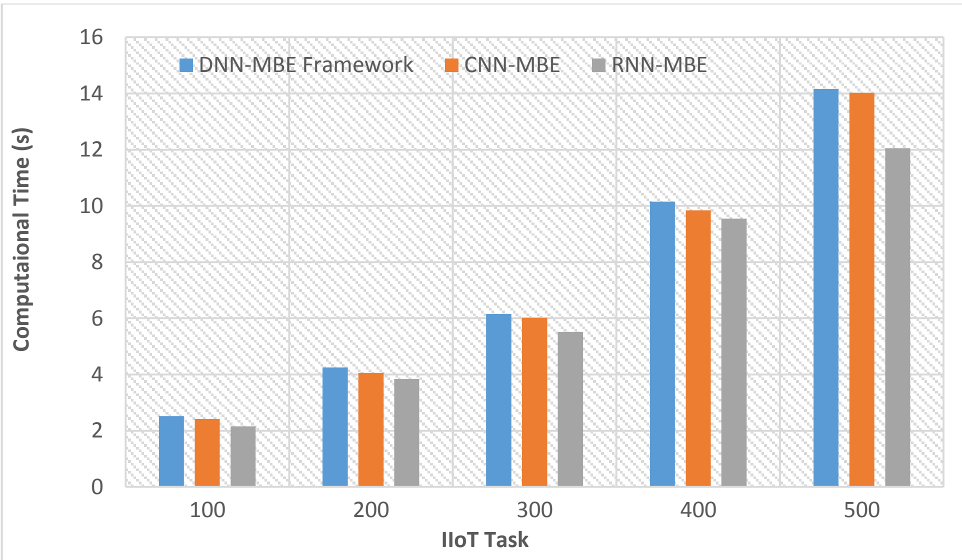

Figure 2a. Computational Time with IoT devices

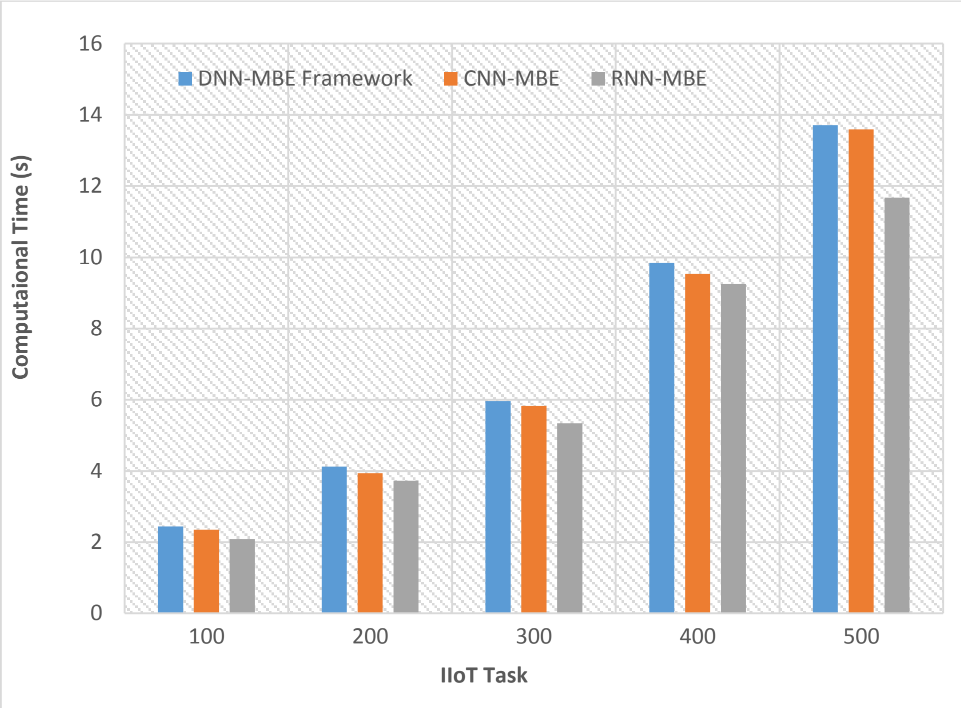

Figure 2b. Computational Time with IIoT devices

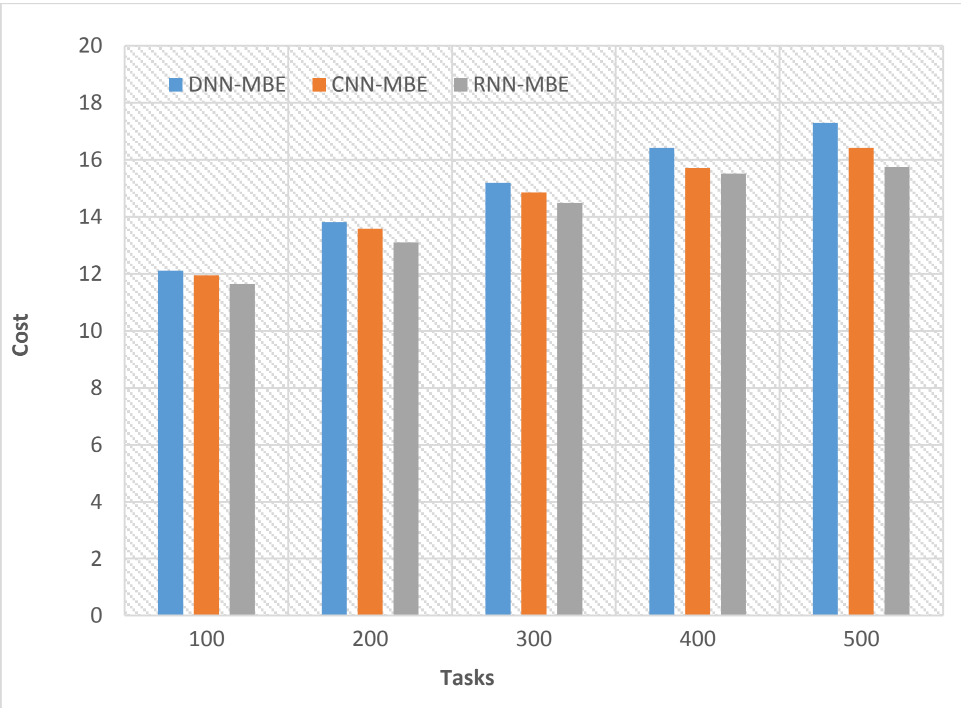

Figure 3a. Task scheduling Cost ($) for IoT devices

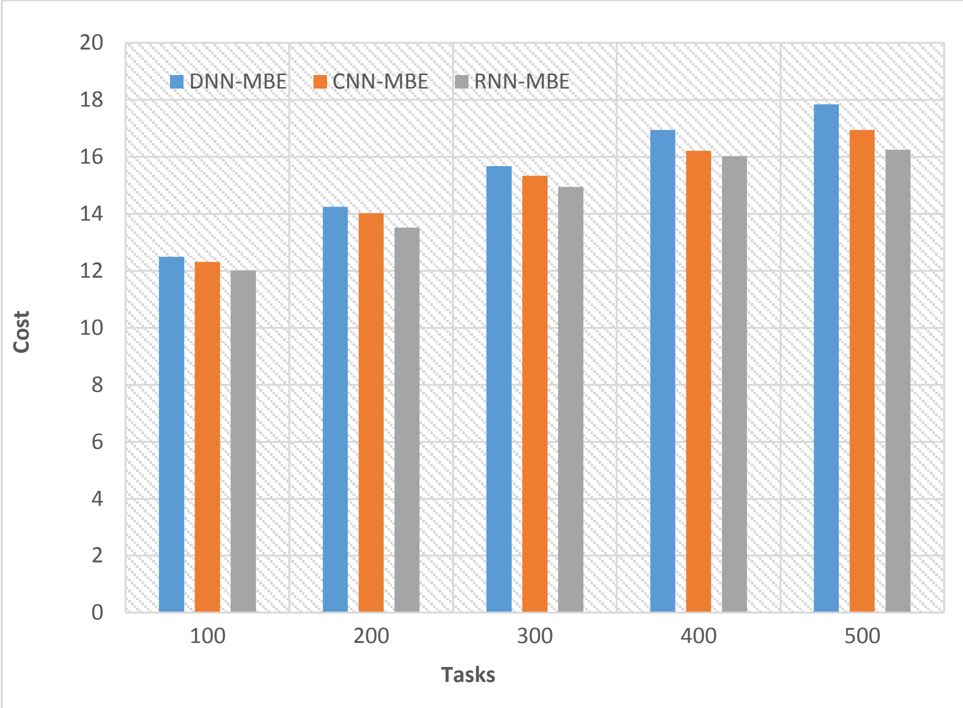

Figure 3b. Task scheduling Cost ($) for IIoT devices

Conclusion

In this research, RAMI 4.0 is integrated with SPES, and this integration of MBE with RAMI uses an RNN. The RNN-MBE, which optimizes the entire process, is responsible for optimizing the process and reducing industry costs. The optimization problem has been fixed, and the MBE simulation has shown that using the proposed MBE is efficient. The simulation result shows that the proposed method achieves a higher rate of scalability in integrating this system with reduced cost and computational complexity. A specialized process known as distributed deep learning (DL) can be helpful in large-scale distributed deep learning (DL) activities that involve extensive training datasets and lengthy training periods. Multiple computational resources may be dedicated to completing a single operation to maximize efficiency. It is possible for numerous distributed nodes in a distributed DL to perform data gathering, data mining, and testing all at the same time and quickly.

Consequently, distributed deep learning is currently considered an outstanding option for implementation within the Industrial Internet of Things. Nevertheless, putting it into practice in a clever and advanced industrial context remains a challenging problem. The most critical issue is finding an efficient solution to manage all the distributed computing resources.

Conflict of interest

The authors declare no potential conflict of interest regarding the publication of this work. In addition, the ethical issues including plagiarism, informed consent, misconduct, data fabrication and, or falsification, double publication and, or submission, and redundancy have been completely witnessed by the authors.

Funding

The author(s) received no financial support for the research, authorship, and/or publication of this article

References

Almadani, B., & Mostafa, S. M. (2021). IIoT based multimodal communication model for agriculture and agro-industries. IEEE Access, 9, 10070-10088.

Ambika, P. (2020). Machine learning and deep learning algorithms on the Industrial Internet of Things (IIoT). Advances in computers, 117(1), 321-338.

Balachander, K., Venkatesan, C., & Kumar, R. (2021). Safety driven intelligent autonomous vehicle for smart cities using IoT. International Journal of Pervasive Computing and Communications.

Binder, C., Draxler, D., Neureiter, C., & Lastro, G. (2020, June). Using a model-based engineering approach for developing Industrial Internet of Things applications. In 2020 IEEE Conference on Industrial Cyberphysical Systems (ICPS) (Vol. 1, pp. 35-40). IEEE.

Binder, C., Neureiter, C., & Lüder, A. (2021). Towards a domain-specific approach enabling tool-supported model-based systems engineering of complex industrial internet-of-things applications. Systems, 9(2), 21.

Boulila, W., Sellami, M., Driss, M., Al-Sarem, M., Safaei, M., & Ghaleb, F. A. (2021). RS-DCNN: A novel distributed convolutional-neural-networks based-approach for big remote-sensing image classification. Computers and Electronics in Agriculture, 182, 106014.

Brecher, C., Buchsbaum, M., Müller, A., Schilling, K., Obdenbusch, M., Staudacher, S., & Albasatineh, M. C. (2021, May). Gaining IIoT insights by leveraging ontology-based modelling of raw data and Digital Shadows. In 2021 4th IEEE International Conference on Industrial Cyber-Physical Systems (ICPS) (pp. 231-236). IEEE.

Khalil, R. A., Saeed, N., Masood, M., Fard, Y. M., Alouini, M. S., & Al-Naffouri, T. Y. (2021). Deep learning in the industrial Internet of things: Potentials, challenges, and emerging applications. IEEE Internet of Things Journal, 8(14), 11016-11040.

Kutuzov, D., Osovsky, A., Stukach, O., & Starov, D. (2021, May). Modeling of IIoT traffic processing by intra-chip NoC routers of 5G/6G networks. In 2021 international siberian conference on control and communications (SIBCON) (pp. 1-5). IEEE.

Ma, X., Yao, T., Hu, M., Dong, Y., Liu, W., Wang, F., & Liu, J. (2019). A survey on deep learning empowered IoT applications. IEEE Access, 7, 181721-181732.

Mantravadi, S., Møller, C., Chen, L. I., & Schnyder, R. (2022). Design choices for next-generation IIoT-connected MES/MOM: An empirical study on smart factories. Robotics and Computer-Integrated Manufacturing, 73, 102225.

Saleem, T. J., & Chishti, M. A. (2021). Deep learning for the Internet of things: Potential benefits and use-cases. Digital Communications and Networks, 7(4), 526-542.

Singh, A. K., Chatterjee, K., & Singh, A. (2022). An Image Security Model Based on Chaos and DNA Cryptography for IIoT Images. IEEE Transactions on Industrial Informatics.

Suthar, K., & He, Q. P. (2021). Multiclass moisture classification in woodchips using IIoT Wi-Fi and machine learning techniques. Computers & Chemical Engineering, 154, 107445.

Zhang, P., Hong, Y., Kumar, N., Alazab, M., Alshehri, M. D., & Jiang, C. (2021). BC-EdgeFL: Defensive Transmission Model Based on Blockchain Assisted Reinforced Federated Learning in IIoT Environment. IEEE Transactions on Industrial Informatics.

Bibliographic information of this paper for citing:

Senthilkumar, P. & Rajesh, K. (2023). Integrated Model-Based Engineering using Deep Learning with IIoT for Industry 4.0. *Journal of Information Technology Management*, 15 (Special Issue), 112-123. https://doi.org/ 10.22059/jitm.2023.91571