*International Journal on Recent and Innovation Trends in Computing and Communication*
*ISSN: 2321-8169 Volume: 11 Issue: 10s*
*DOI: https://doi.org/10.17762/ijritcc.v11i10s.7691*
***Article Received:** 07 June 2023 **Revised:** 25 July 2023 **Accepted:** 16 August 2023*

# Dynamic Resource Allocation in Industrial Internet of Things (IIoT) using Machine Learning Approaches

**Pankaj Singh Sisodiya[1], Dr. Vijay Bhandari[2]**
[1]Research Scholar, Department of Computer Science & Engineering
[2]Associate Professor, Department of Computer Science & Engineering
[1,2]Madhyanchal Professional University, Bhopal, India.
sisodiya.pankaj90@gmail.com, bhandarivijay314@gmail.com

**Abstract:** In today's era of rapid smart equipment development and the Industrial Revolution, the application scenarios for Internet of Things (IoT) technology are expanding widely. The combination of IoT and industrial manufacturing systems gives rise to the Industrial IoT (IIoT). However, due to resource limitations such as computational units and battery capacity in IIoT devices (IIEs), it is crucial to execute computationally intensive tasks efficiently. The dynamic and continuous generation of tasks poses a significant challenge to managing the limited resources in the IIoT environment. This paper proposes a collaborative approach for optimal offloading and resource allocation of highly sensitive industrial IoT tasks. Firstly, the computation-intensive IIoT tasks are transformed into a directed acyclic graph. Then, task offloading is treated as an optimization problem, taking into account the models of processor resources and energy consumption for the offloading scheme. Lastly, a dynamic resource allocation approach is introduced to allocate computing resources to the edge-cloud server for the execution of computation-intensive tasks. The proposed joint offloading and scheduling (JOS) algorithm creates its DAG and prepare a offloading queue. This queue is designed using collaborative q-learning based reinforcement learning and allocate optimal resources to the JOS for execution of tasks present in offloading queue. For this machine learning approach is used to predict and allocate resources. The paper compares conventional and machine learning-based resource allocation methods. The machine learning approach performs better in terms of response time, delay, and energy consumption. The proposed algorithm shows that energy usage increases with task size, and response time increases with the number of users. Among the algorithms compared, JOS has the lowest waiting time, followed by DQN, while Q-learning performs the worst. Based on these findings, the paper recommends adopting the machine learning approach, specifically the JOS algorithm, for joint offloading and resource allocation.

**Keywords:** Internet of Things (IoT), Industrial IoT, Task Offloading, Resource Allocation, Optimal, Machine Learning.

## I. Introduction

Every day, your business generates a vast amount of data as it undergoes digitization across its entire value chain, from procurement to production to delivery and service. This digital transformation connects various aspects of your organization, and advancements in digital technologies such as autonomous plant operations, efficient electrical drives, and controls further support these efforts [1][2]. The term "digitalization" in business refers to the innovative use of digital technologies to fundamentally change the business model, unlock new revenue streams, and provide value-added services or products to customers. As a result, Industry 4.0 is revolutionizing how companies manufacture, enhance, and distribute their products [3][4]. Manufacturers are integrating emerging technologies like the Internet of Things (IoT), cloud computing, analytics, and artificial intelligence (AI) and machine learning into their production facilities and overall operations. The significance of digitalization in business became evident during the pandemic, as digitally transformed businesses experienced less disruption and recovered more quickly compared to those that had not embraced digitalization [5]. Industry 4.0, combined with the Industrial Internet of Things (IIoT), equips businesses with advanced sensors, embedded software, and robotics, enabling data collection, analysis, and improved decision-making. The real value emerges when data from production operations is combined with operational data from enterprise systems like ERP, supply chain, and customer service, enabling unprecedented visibility and insights from previously isolated information [6]. Utilizing high-tech IIoT devices in smart factories leads to increased productivity and improved quality. AI-powered visual insights replace traditional manual inspection models, reducing manufacturing errors and saving time and money. With minimal investment, quality control personnel can set up a smartphone connected to the cloud, enabling remote monitoring of manufacturing processes from virtually anywhere [7][8].

Manufacturers can utilize machine learning algorithms to promptly detect errors, thereby avoiding costly repairs at later stages. The concepts and technologies of Industry 4.0 are applicable to various industrial sectors, including discrete and process manufacturing, oil and gas, mining, and others. The emergence of the Industrial Internet of Things (IoT) has led to an increased volume of information generated through the connection between people and smart objects. This connectivity

*International Journal on Recent and Innovation Trends in Computing and Communication*
***ISSN: 2321-8169 Volume: 11 Issue: 10s***
***DOI: https://doi.org/10.17762/ijritcc.v11i10s.7691***
***Article Received:** 07 June 2023 **Revised:** 25 July 2023 **Accepted:** 16 August 2023*

introduces new challenges in terms of the interaction between humans and connected devices, which can occur anytime and anywhere. One such challenge is resource allocation. While cloud computing and IoT offer virtually limitless processing resources, they also introduce the risk of unacceptable processing delays for time-critical Industrial Internet of Things (IIoT) applications. Moreover, as the volume of generated data continues to surge, different levels of resources are required to handle this data effectively [9][10]. Dynamic resource allocation is a solution that ensures efficient, reliable, and scalable processing of this data. By processing only the necessary data, unnecessary data transfer and storage are reduced, resulting in the optimal utilization of computing resources [11]. Cloud computing enables organizations to access a pool of computing resources via the internet, allowing them to scale their resources based on demand. It facilitates efficient allocation of computing resources by quickly provisioning and de-provisioning resources as needed, without relying on physical hardware.

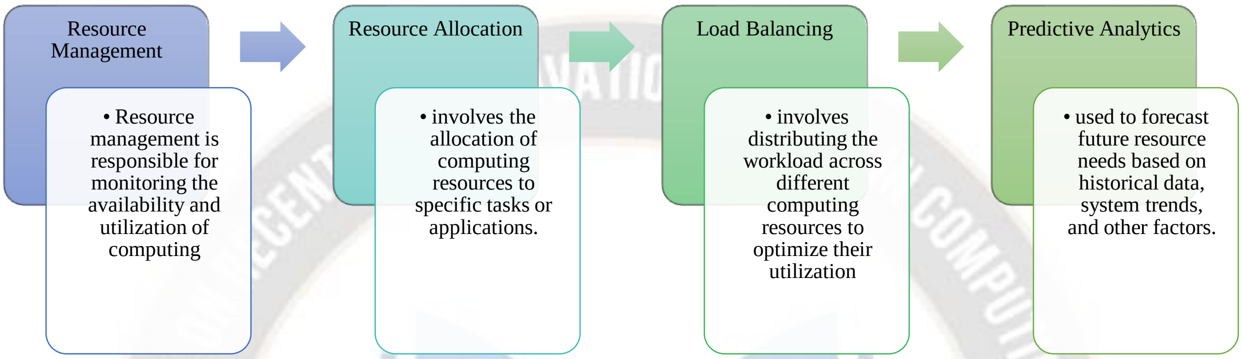



Fig. 1. Components of Dynamic Resource Allocation in IIoT

The IoT ecosystem comprises numerous smart objects, and effective communication and operation of these objects require resource allocation. Resource allocation is necessary for several reasons:

- Ensuring Quality of Service (QoS) by allocating resources appropriately.
- Managing system irregularities to maintain reliability.
- Reducing idle time of devices waiting for resource allocation.
- Enabling dynamic resource scheduling and optimal utilization of available resources.
- Minimizing power consumption.

To achieve these goals, machine learning algorithms can be trained to analyze data, identify patterns, and establish relationships between variables like resource usage, system load, and application performance. This analysis enables predicting future resource requirements and optimizing real-time resource allocation [12][13]. This paper introduces a joint approach for optimal task offloading and dynamic resource allocation. By leveraging machine learning algorithms to analyze data traffic and learn resource usage patterns, this approach enhances energy efficiency, reduces power loss, and minimizes costs. It ensures efficient resource allocation, effectively utilizing energy and minimizing power wastage.

The key features of paper are:

- The paper presents a methodology for optimal offloading and resource allocation in highly sensitive industrial Internet of Things (IIoT) tasks.
- The methodology involves converting tasks into a directed acyclic graph (DAG), formulating offloading as an optimization problem, and implementing a dynamic resource allocation approach using collaborative Q-learning.
- This approach aims to minimize energy consumption while meeting latency and reliability requirements by effectively distributing tasks between edge devices and cloud servers based on workload and task requirements.

The remaining section of the paper is organized as: Section 2 presents the literature review of the resource allocation in industrial IoT, section 3 presents the proposed methodology with algorithm, section 4 presents the results and discussion for proposed methodology. Section 5 presents the conclusion and future work.

## II. Literature Review

The summarized studies focus on enhancing resource management, optimization, and performance in Industrial Internet of Things (IIoT) networks using machine learning, deep reinforcement learning (DRL), and blockchain technologies. Zhang et al. [14] proposed a FL algorithm assisted by DRL for managing data from IIoT equipment, achieving high accuracy. Manogaran et al. [15] introduced a method combining learning-assisted slicing and resource allocation processes to improve service reliability for 6G users. Liu et al. [16] presented a DRL-based framework for optimizing performance in blockchain-enabled IIoT systems. Zhang et al. [17] developed a deep reinforcement learning-

*International Journal on Recent and Innovation Trends in Computing and Communication*
*ISSN: 2321-8169 Volume: 11 Issue: 10s*
*DOI: https://doi.org/10.17762/ijritcc.v11i10s.7691*
***Article Received:** 07 June 2023 **Revised:** 25 July 2023 **Accepted:** 16 August 2023*

based resource management scheme for real-time optimal resource allocation decisions in DNN inference. Shi et al. [18] addressed spectrum resource management challenges in the IIoT network using a Modified Deep Q-learning Network (MDQN). Fan et al. [19] proposed a joint task offloading and resource allocation scheme for accuracy-aware machine learning-based IIoT applications. Tang et al. [20] proposed a multi-exit-based Federated Edge Learning (ME-FEEL) framework to enable maximum device participation in IIoT networks with limited resources. Bommisetty et al. [21] presented a Phasic Policy Gradient (PPG) based Time-Slotted Channel Hopping (TSCH) schedule learning algorithm for improving throughput and energy efficiency in TSCH networks. Yang et al. [22] introduced an optimization framework for blockchain-enabled IIoT systems to reduce energy consumption. These studies collectively contribute to advancing IIoT network management and optimization using innovative technologies. Rosenberger et al. [23] applied deep reinforcement learning (DRL) for intelligent resource allocation in Industrial Internet of Things (IIoT) devices. The study demonstrated low resource usage and transferable learned behavior. Boobalan et al. [24] discussed the integration of Federated Learning (FL) with IIoT to ensure data privacy. They provided an overview of privacy, resource, and data management, and highlighted potential applications and future research directions. Tang et al. [25] proposed a deep reinforcement learning-based algorithm for resource allocation in Satellite Internet of Things (S-IoT). The algorithm balanced resource efficiency, QoS requirements, and power control, with promising results shown in simulations. Li et al. [26] addressed resource management challenges in mobile-edge computing (MEC) for IoT. They proposed a cloud-edge collaborative resource allocation framework using 6G and blockchain technology, and introduced a collective reinforcement learning (CRL) method for intelligent resource allocation. Fang et al. [28] presented a deep reinforcement learning-based resource allocation scheme for content distribution in a fog radio access network (FRAN). They utilized in-network caching and cloud-edge cooperation to improve resource utilization and content delivery. Jayalaxmi et al. [29] introduced DeBot, a deep learning model for bot detection in industrial network traffic. They utilized a cascade forward backpropagation neural network (CFBPNN) and feature selection techniques, achieving high accuracy in bot detection. Yang et al. [30] proposed a clustering-based sharded blockchain strategy for collaborative computing in the IoT. They used K-means clustering and deep reinforcement learning to optimize user grouping, consensus nodes, and scalability of the sharded blockchain. Gong et al. [30] addressed resource allocation challenges in Low Power Wide Area Networks (LPWAN), specifically LoRaWAN. They proposed a dynamic reinforcement learning resource allocation (DRLRA) approach to improve performance in terms of consumption and reliability. Olatinwo et al. [31] conducted a bibliometric analysis and comprehensive review of resource management in IoT networks. They identified research challenges and opportunities, highlighting the growing importance of deep learning approaches for resource management. Tian et al. [32] investigated offloading decisions, CPU frequencies, and transmit powers optimization in multi-Mobile Edge Computing (MEC) and multi-IoT device networks. They developed a deep reinforcement learning-based algorithm to minimize computing pressure, energy consumption, and task dropping cost. Lei et al. [33] proposed an adaptive resource allocation method for Industrial Internet of Things (IIoT) nodes. The method evaluates the self-similarity of observation data to prioritize nodes dynamically and uses the Deep Q Network (DQN) algorithm.

## III. Proposed Methodology

In this paper, a methodology for addressing the challenge of optimal offloading and resource allocation for highly sensitive industrial Internet of Things (IIoT) tasks. The goal is to efficiently distribute computational tasks between edge devices and cloud servers to maximize system performance and minimize resource consumption. To achieve this, the methodology adopted a collaborative approach that involves multiple components. In first step, the computation-intensive IIoT tasks are converted into a directed acyclic graph (DAG). This representation helps capture the dependencies and relationships between different tasks, allowing for better analysis and optimization. Next step formulate the task offloading as an optimization problem. They develop models that describe the available processor resources and energy consumption associated with the task offloading scheme. By considering these factors, they aim to find the optimal allocation strategy that minimizes energy consumption while meeting the latency and reliability requirements of the IIoT tasks. Finally, in last step a dynamic resource allocation approach is presented. This approach aims to allocate computing resources from the edge-cloud server to the computation-intensive tasks effectively. By dynamically adjusting the resource allocation based on the workload and task requirements, the system can efficiently utilize available resources and adapt to changing conditions.

### 3.1 System Model

The system is modelled as edge-cloud server for providing resources to industrial IoT networks, as presented in fig 2.

*International Journal on Recent and Innovation Trends in Computing and Communication*
***ISSN: 2321-8169 Volume: 11 Issue: 10s***
***DOI: https://doi.org/10.17762/ijritcc.v11i10s.7691***
***Article Received:** 07 June 2023 **Revised:** 25 July 2023 **Accepted:** 16 August 2023*

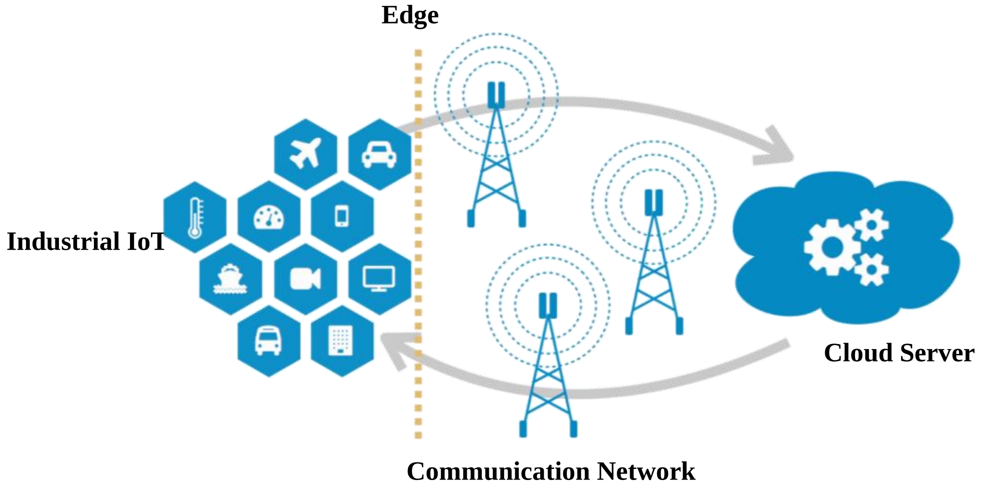



Fig. 2. Resource Allocation Support to IIoT with Edge-Cloud Server

The entire network framework is composed of task processing layer (IIoT layer), Server layer (Edge-Cloud). In task processing layer, all task related informations are collected such as task completion time (TCT), task length (TL) and task size (TS). According to this information, queue is maintained to be migrated. Whereas server layer provides all necessary information related with processing units, storage, data transfer capacity, speed, for virtual machines with respective host servers. These task layer's information as well as datacenter layer's information is helpful for designing best scheduling algorithms for obtaining optimal solution.The paper aims to schedule a set of tasks according to available virtual machines (VMs) in the cloud to minimize cost, latency and energy. The schematic flow of proposed system is presented below in fig 3.

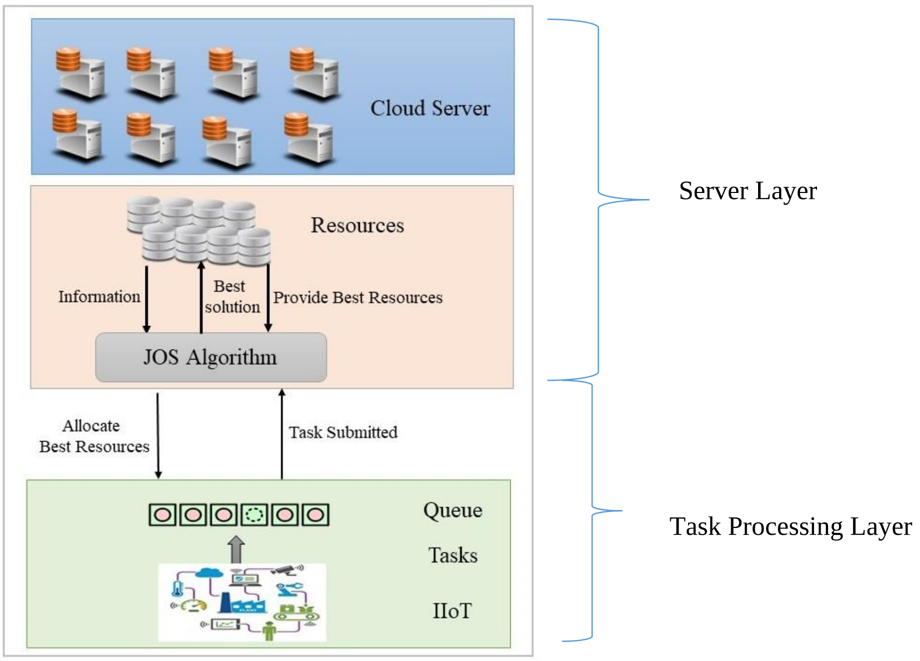



Fig. 3. General Flow of Joint Offloading and Resource Allocation in Edge-Cloud Server for IIoT

**Task Processing Layer:** The Task Processing Layer (TPL) manages the workload of IIoT devices or tasks using a Directed Acyclic Graph (DAG) representation. The DAG consists of a set of vertices (V) and edges (E), denoted as $G_{DAG} = \{V, E\}$. Each vertex represents a task, and the edges represent the dependency constraints between tasks and the offloading queue. Task $T_j$ can only be executed after task $T_i$ is completed. Tasks that can be executed in parallel are placed at the same level in the graph. For example, if among 10 tasks, $T_1, T_5, and\ T_8$ can be executed in parallel, they will be placed at the same level. Workflow scheduling is considered a

*International Journal on Recent and Innovation Trends in Computing and Communication*
***ISSN: 2321-8169 Volume: 11 Issue: 10s***
***DOI: https://doi.org/10.17762/ijritcc.v11i10s.7691***
***Article Received:** 07 June 2023 **Revised:** 25 July 2023 **Accepted:** 16 August 2023*

scheduling problem in order to minimize execution time while ensuring a balanced overall load.

**Server Layer:** The server layer (SL) presents an edge-cloud infrastructure using multiple instances of virtual machines ($VM$), each with its specific characteristics. Each $VM$ ($i^{th}$ $VM$) is characterized by its processing capacity ($VM_{i_c}$), data transfer rate ($VM_{i_{dr}}$), and bandwidth ($VM_{i_b}$). These characteristics are represented as random variables. In addition to these characteristics, cost optimization is also considered as one of the crucial factors for server performance.

### 3.2 Joint Offloading and Scheduling Algorithm

The proposed joint offloading and scheduling (JOS) algorithm model is composed of two steps (Fig 4): (i) Optimal DAG creation and offloading queue generation, (ii) workflow load prediction and dynamic scheduling.

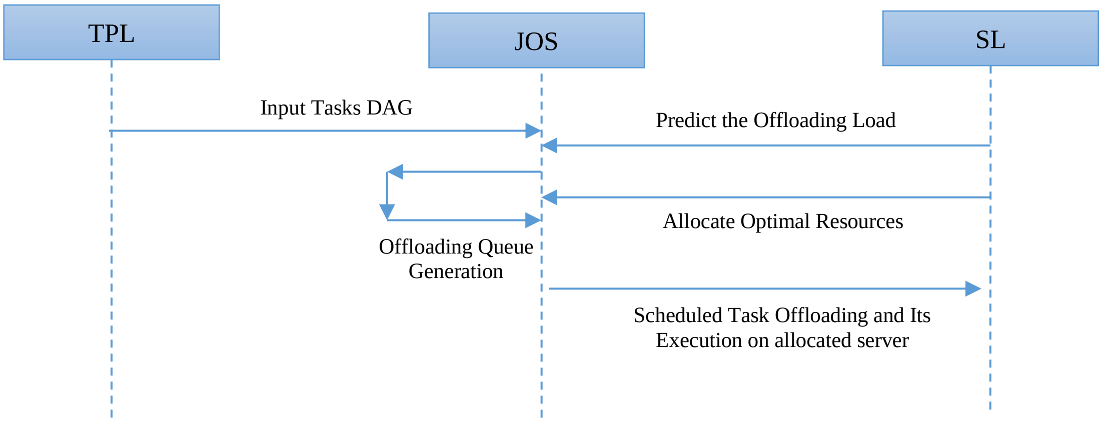



Fig. 4. Schematic Flow of Proposed JOS Algorithm

When an IIoT task arrives, the Task Processing Layer (TPL) extracts the relevant features from the task. These features capture important information about the task, such as its computational requirements, dependencies, and priority. The Based on the task features, the Job Offloading System (JOS) generates a Directed Acyclic Graph (DAG). The DAG describes the dependencies and relationships between tasks and aids in task execution order determination. JOS creates an offloading queue, which is an ordered list of jobs sorted by priority. Prioritization is determined via collaborative q-learning-based reinforcement learning, with aspects such as task urgency, resource availability, and system efficiency considered. Section 3.3 describes the collaborative q-learning. The traffic information acquired from the offloading queue is provided by JOS to the Resource Scheduler Layer (SL). This data assists SL in forecasting system demand and making appropriate resource allocation decisions. SL analyses received traffic data and using machine learning methods to forecast system demand. These algorithms may estimate the system's resource requirements based on historical data, present system status, and other factors. SL finds the appropriate allocation of resources for executing the jobs in the offloading queue based on the load forecast. This allocation takes into account aspects including available resources, task priority, and system limits. The approach for resource allocation seeks to maximise system performance and efficiency.

Based on their priority, the algorithm iterates over the jobs in the offloading queue. To execute each task, the allotted resources from SL are assigned. Allocating computing resources such as CPU, memory, and network bandwidth falls under this category. The work is completed with the resources allotted. The specific execution mechanism is determined by the task's nature and the system architecture. Depending on the offloading decisions made by JOS, it could involve running the task locally or offloading it to remote resources. During task execution, the system monitors the tasks' progress and performance. Any system status updates, such as resource utilisation or job completion, are logged. This enables the system to handle a constant stream of tasks, modifying its resource allocation and task execution procedures in response to the dynamically changing workload and system conditions. Algorithm 1 contains the whole algorithm. As a result, the proposed Joint offloading and resource allocation algorithms in dynamic IIoT contexts offer several benefits. They boost system performance by optimising job distribution and resource allocation, ensuring that current resources are exploited to their full potential. Load balancing works by dynamically splitting duties between local devices and the cloud, minimising resource bottlenecks. By reducing energy consumption at local devices, offloading work to the cloud improves energy efficiency. The solutions are scalable and flexible to shifting workloads and needs. They also reduce latency by utilising cloud resources to expedite task execution. Additionally, cost optimization is achieved by reducing infrastructure and maintenance costs and utilizing cost-effective cloud resources.

*International Journal on Recent and Innovation Trends in Computing and Communication*
***ISSN: 2321-8169 Volume: 11 Issue: 10s***
***DOI: https://doi.org/10.17762/ijritcc.v11i10s.7691***
***Article Received:** 07 June 2023 **Revised:** 25 July 2023 **Accepted:** 16 August 2023*

Algorithm 1: *Joint Offloading and Scheduling Algorithm*

1: Initialize System
2: When IIoT tasks arrive
3: TaskFeatures←TPL.extractTaskFeatures(IIoTTask)
4: JOS.createDAG(TaskFeatures)
5: OffloadingQueue ←JOS.prepareOffloadingQueue(collaborativeQTable)
6: TrafficInfo ← JOS.getTrafficInformation(OffloadingQueue)
7: LoadPrediction ←SL.predictLoad(TrafficInfo)
8: ResourceAllocation ← SL.allocateResources(LoadPrediction)
9: for each Task in OffloadingQueue
    ExecuteTask(Task, ResourceAllocation)
    ExecuteTask(Task, ResourceAllocation)
    AllocateResources(Task, ResourceAllocation)
    Execute(Task)
    UpdateSystemStatus(Task)
10: Repeat the process for subsequent IIoT task arrivals

### 3.3 Collaborative Q-Learning for JOS Algorithm

Collaborative Q-learning is a type of reinforcement learning designed with numerous agents that collaborate to learn an optimal policy within shared environment. The base of this model is Q-learning algorithm for reinforcement learning. In conventional Q-learning single agent cooperates with the shared environment and train itself to make optimal decisions by updating a Q-value table. Q-value table determines the decision based on past certain events. However, in proposed collaborative Q-learning approach, multiple agents are deployed to work together and perform learning as a joint policy that maximises the total rewards for the decision. In collaborative Q-learning, each agent makes their own state-action pair and generate their respective Q-values. Then their decisions are fused together and their collaborative results are termed as final Q-value. The basic idea behind collaborative Q-learning is to extend the Q-learning algorithm to handle the complexities of several agents. Each agent is equipped with a Q-value table that converts state-action pairs to Q-values. Unlike single-agent Q-learning, however, Q-value updates take into account not only the agent's own experiences, but also the experiences of other agents in the team. Collaboration in Q-learning can take place in a variety of ways. Agents sharing their Q-values with one another is a popular strategy. Another alternative is to use a centralised Q-value function. Instead of each agent maintaining its own Q-value table, all potential state-action pairings are stored in a centralised repository. Agents can then access and update this centralised Q-value function based on their own experiences. This strategy allows for more efficient agent coordination and communication. Collaborative Q-learning can be used to a wide range of multi-agent scenarios, including cooperative tasks in which agents work together to achieve a common goal and competitive scenarios in which agents compete against one another. The goal is to create a collaborative policy that maximises the overall performance of the team.

Algorithm 1: *Collaborative Q-Learning*

1: Initialize
 - Q-value tables for each agent $Q1, Q2, \ldots, Qn$
 - Shared environment $E$
 - Hyperparameters: learning rate alpha, discount factor gamma, exploration rate epsilon
2: Repeat the following steps for each episode:
  Reset the environment $E$
  Initialize the state $S$
3: Repeat the following steps until the episode terminates:
4: For each agent i = 1 to $n$
   select the action a with the maximum Q-value for the current state from $Q_i$
   Take action a, observe the next state S', and receive the reward R
   Update the Q-value of the current state-action pair $Q_i$ (S, a):
   $Q_i$ (S, a) = (1 - alpha) * $Q_i$ (S, a) + alpha * (R + gamma * max($Q_i$ (S', a')))
   Update the state S to S'
   If any termination conditions are met:
   Break the loop
5: Reduce epsilon according to the exploration rate decay schedule
6: Return the learned Q-value tables $Q1, Q2, \ldots, Qn$

## IV. Results and Discussion

To evaluate the effectiveness of the JOS, the proposed algorithm is implemented using Matlab R-2020a. The simulations were conducted on an Intel i5, 3.7Ghz PC with 8 GB RAM.In this work, node-server computing system consisting of IIoT tasks and multiple edge-cloud servers is designed. The parameters of the simulation are shown in table 1.

Table 1. System Parameters Used

| Parameters | Value |
|---|---|
| Maximum Number of IIoT devices | 500 |
| Maximum Number of servers | 50 |
| Bandwidth of edge server | 12MB/s |
| Bandwidth of IoT nodes | 300MB/s |
| Data Size | 250kb-1Mb |
| Ideal delay of request | 0.4-0.6s |

In this research work, following parameters are used:

*International Journal on Recent and Innovation Trends in Computing and Communication*
***ISSN: 2321-8169 Volume: 11 Issue: 10s***
***DOI: https://doi.org/10.17762/ijritcc.v11i10s.7691***
***Article Received:** 07 June 2023 **Revised:** 25 July 2023 **Accepted:** 16 August 2023*

Average Response Time: Response time of resource allocation refers to the time it takes for a system or process to allocate resources to a particular task or request. It can be defined mathematically as follows:

$$ART = \frac{|Alloc_{time} - Arrival_{time}|}{N}$$

Where, $Alloc_{time}$ = Allocation time

$Arrival_{time}$ = Arrival time

N = Number of tasks

Average Delay: Time taken to allocate resources and execute all task is termed as average delay. It can be defined mathematically as follows:

$$AD = \frac{|Exe_{time} - Alloc_{time}|}{N}$$

Where, $Alloc_{time}$ = Resource Allocation time

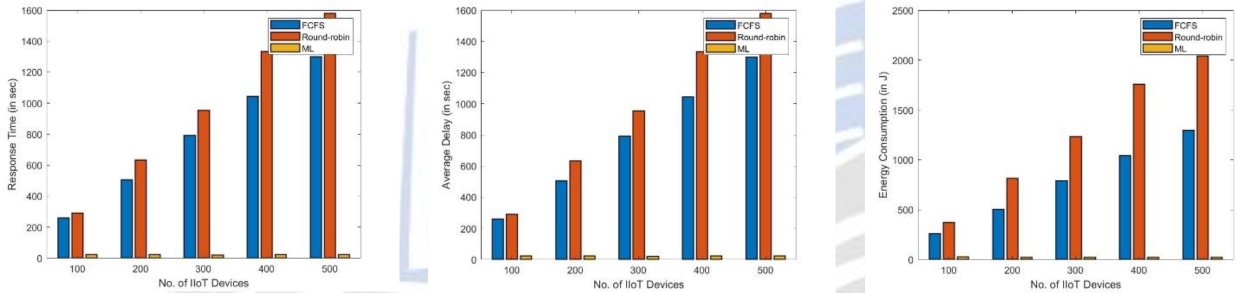
$Exe_{time}$ = Execution time

N = Number of tasks

Average Energy Consumption: Total amount of energy required to execute all tasks. It can be defined mathematically as follows:

$$AEC = \frac{Energy\ required\ to\ process\ n - bits}{N}$$

Below in fig 5, the paper presents the result analysis of the conventional approach based dynamic resource allocation versus machine learning based resource allocation. In fig 5 (a) average response time graph is plotted with respect to varying IIoT devices. Similarly, fig 5(b) presents the average delay occurred during decision making is plotted with respect to varying IIoT devices. In fig 5(c), energy consumption is plotted with respect to varying IIoT devices. From these graphs it is clearly visible that machine learning approach outperforms better as compared to conventional approach in terms of all relevent parameters. Therefore, we have adopted machine learning approach for providing joint offloading and resource allocation benefits to the server.




(a) Response Time (b) Average Delay (c) Energy Consumption

Fig 5. Comparison of Conventional Approach Versus Machine Learning based Dynamic Resource Allocation

In this section, we have presented a result analysis for two cases. One for variable number of IoT nodes/users and another for variable size of tasks. The result was evaluated in terms of response time as well as energy used. Fig 6 shows the energy usage of proposed algorithm with variable task size. In this case number of IoT nodes/users are fixed and task size is varied. The task size was taken from 250Kb to 1000Kb. The result analysis shows that with increases size of task energy utilization also increases. Fig 7 shows the response time of proposed algorithm with variable number of users. In this case number of IIoT devices are varied and task size is fixed. The number of IIoT devices was taken from 100-500. The result analysis shows that with increases size of users, response time also increases.

*International Journal on Recent and Innovation Trends in Computing and Communication*
***ISSN: 2321-8169 Volume: 11 Issue: 10s***
***DOI: https://doi.org/10.17762/ijritcc.v11i10s.7691***
***Article Received:** 07 June 2023 **Revised:** 25 July 2023 **Accepted:** 16 August 2023*

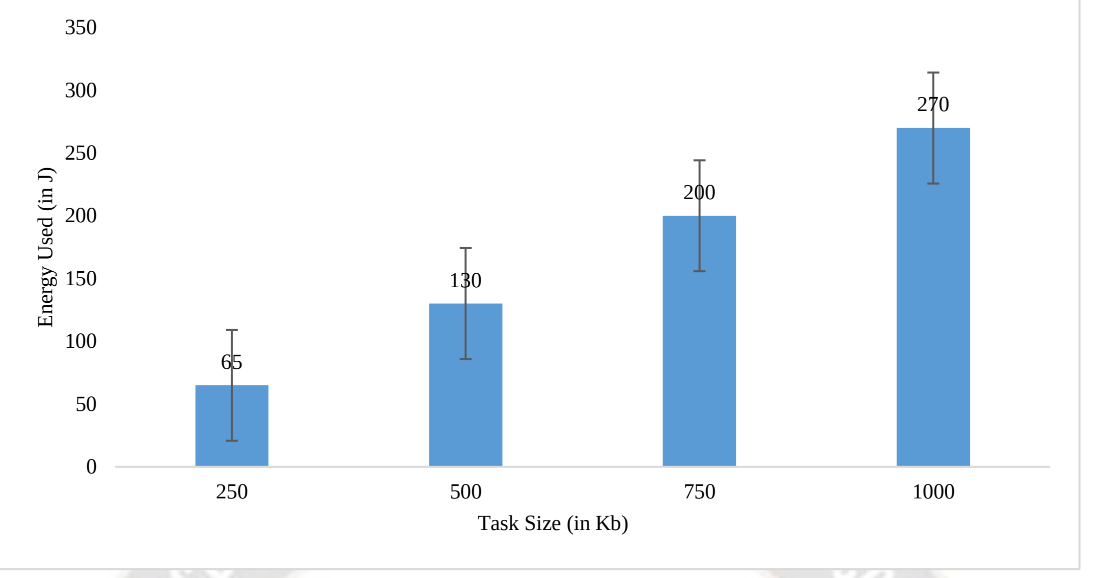



Fig 6. Energy Usage with Variable Task Size

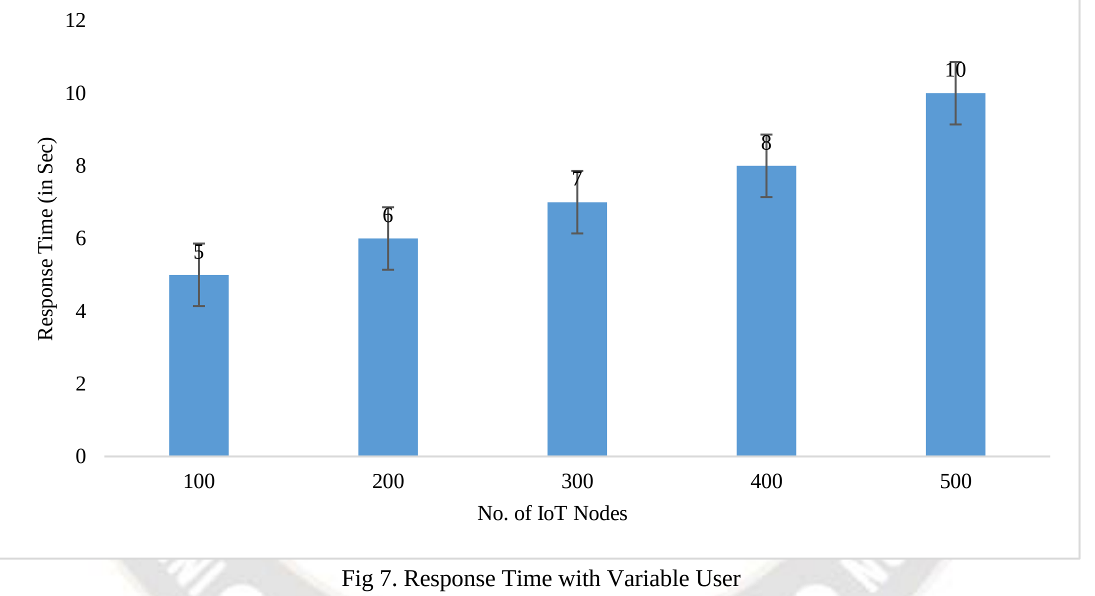



Fig 7. Response Time with Variable User

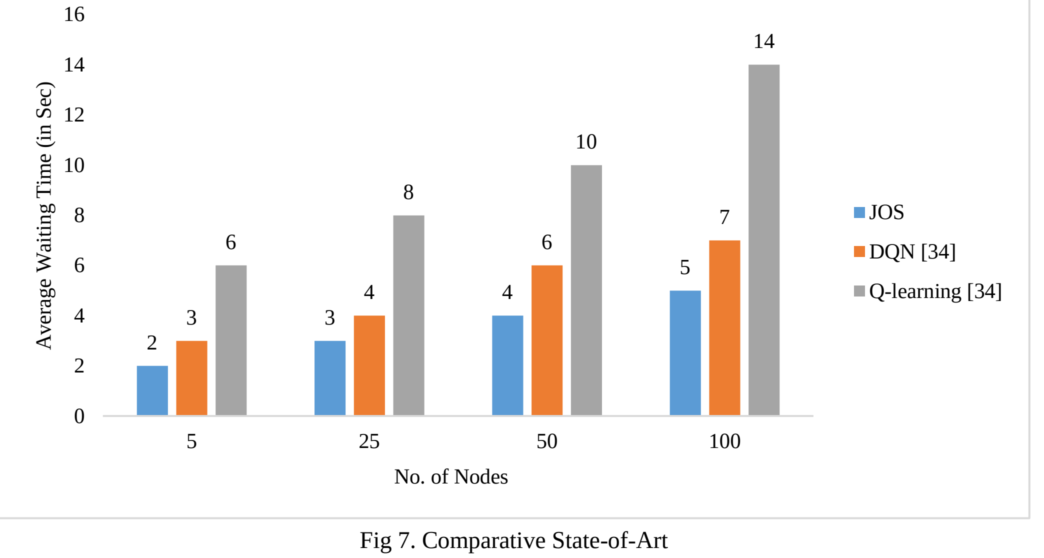



Fig 7. Comparative State-of-Art

*International Journal on Recent and Innovation Trends in Computing and Communication*
***ISSN: 2321-8169 Volume: 11 Issue: 10s***
***DOI: https://doi.org/10.17762/ijritcc.v11i10s.7691***
***Article Received:** 07 June 2023 **Revised:** 25 July 2023 **Accepted:** 16 August 2023*

Based on fig 7, it appears that the values for JOS, DQN, and Q-learning increase as the number of node increases. This could suggest that the performance or effectiveness of these algorithms improves with a larger number of nodes. Here are some possible inferences based on the given data:

- JOS seems to have the lowest average waiting time (AWT) as compared to DQN and Q-learning.
- DQN consistently has higher AWT than JOS but lower than Q-learning with varying nodes.
- Q-learning has the highest values among the three algorithms in every scenario, indicating that it exhibits the worst performance or achieves the highest AWT whereas proposed JOS have lowest performance.

## V. Conclusion

In recent years, the integration of IoT devices in networking applications has become increasingly common. However, managing and monitoring these networks has become challenging due to the large number of devices, complex systems, and high data volumes generated. To address these difficulties, researchers have explored the use of machine learning techniques to develop dynamic decision-making mechanisms for IoT systems, considering the incomplete information available about the environment. In this context, the methodology presented in the paper provides a comprehensive framework for addressing the challenges of offloading and resource allocation in highly sensitive industrial IoT environments. The algorithm aims to optimize the overall system performance by intelligently offloading tasks, predicting the system load, and allocating resources efficiently. The collaborative q-learning and machine learning approaches help in making informed decisions regarding task prioritization, load prediction, and resource allocation, ensuring efficient utilization of available resources and timely execution of IIoT tasks. According to results presented, the proposed approach effectively reduces the average delay of tasks, leading to improved performance in IoT systems. The result compares conventional resource allocation with machine learning-based resource allocation in terms of response time, delay, and energy consumption. The proposed algorithm's energy usage increases with the size of the task, while the response time increases with the number of users. Among the algorithms compared (JOS, DQN, and Q-learning), JOS has the lowest average waiting time, followed by DQN, and Q-learning performs the worst. Based on these findings, the paper recommends adopting the machine learning approach, specifically the JOS algorithm, for joint offloading and resource allocation.

## References

[1] Sangaiah AK, Hosseinabadi AAR, Shareh MB, Bozorgi Rad SY, Zolfagharian A, Chilamkurti N. IoT Resource Allocation and Optimization Based on Heuristic Algorithm. Sensors (Basel). 2020 Jan 18;20(2):539. doi: 10.3390/s20020539. PMID: 31963762; PMCID: PMC7014514.

[2] S. Vimal, Manju Khari, Nilanjan Dey, Rubén González Crespo, Y. Harold Robinson, Enhanced resource allocation in mobile edge computing using reinforcement learning based MOACO algorithm for IIOT, Computer Communications, Volume 151, 2020, Pages 355-364, ISSN 0140-3664, https://doi.org/10.1016/j.comcom.2020.01.018.

[3] P. Goswami, A. Mukherjee, M. Maiti, S. K. S. Tyagi and L. Yang, "A Neural-Network-Based Optimal Resource Allocation Method for Secure IIoT Network," in IEEE Internet of Things Journal, vol. 9, no. 4, pp. 2538-2544, 15 Feb.15, 2022, doi: 10.1109/JIOT.2021.3084636.

[4] Y. Chen, Z. Liu, Y. Zhang, Y. Wu, X. Chen and L. Zhao, "Deep Reinforcement Learning-Based Dynamic Resource Management for Mobile Edge Computing in Industrial Internet of Things," in IEEE Transactions on Industrial Informatics, vol. 17, no. 7, pp. 4925-4934, July 2021, doi: 10.1109/TII.2020.3028963.

[5] Jamil, Sonain, MuhibUr Rahman, and Fawad. 2022. "A Comprehensive Survey of Digital Twins and Federated Learning for Industrial Internet of Things (IIoT), Internet of Vehicles (IoV) and Internet of Drones (IoD)" *Applied System Innovation* 5, no. 3: 56. https://doi.org/10.3390/asi5030056

[6] P. Ambika, Chapter Thirteen - Machine learning and deep learning algorithms on the Industrial Internet of Things (IIoT), Editor(s): Pethuru Raj, Preetha Evangeline, Advances in Computers, Elsevier, Volume 117, Issue 1, 2020, Pages 321-338, ISSN 0065-2458, ISBN 9780128187562, https://doi.org/10.1016/bs.adcom.2019.10.007.

[7] Mohamed Said Frikha, Sonia Mettali Gammar, Abdelkader Lahmadi, Laurent Andrey, Reinforcement and deep reinforcement learning for wireless Internet of Things: A survey, Computer Communications, Volume 178, 2021, Pages 98-113, ISSN 0140-3664, https://doi.org/10.1016/j.comcom.2021.07.014.

[8] W. Sun, J. Liu and Y. Yue, "AI-Enhanced Offloading in Edge Computing: When Machine Learning Meets Industrial IoT," in IEEE Network, vol. 33, no. 5, pp. 68-74, Sept.-Oct. 2019, doi: 10.1109/MNET.001.1800510.

[9] B. Chen, J. Wan, Y. Lan, M. Imran, D. Li and N. Guizani, "Improving Cognitive Ability of Edge Intelligent IIoT through Machine Learning," in IEEE Network, vol. 33, no. 5, pp. 61-67, Sept.-Oct. 2019, doi: 10.1109/MNET.001.1800505.

[10] X. Xu, R. Mo, F. Dai, W. Lin, S. Wan and W. Dou, "Dynamic Resource Provisioning With Fault Tolerance for Data-Intensive Meteorological Workflows in Cloud," in IEEE Transactions on Industrial Informatics, vol. 16, no. 9, pp. 6172-6181, Sept. 2020, doi: 10.1109/TII.2019.2959258.

[11] H. Liang et al., "Reinforcement Learning Enabled Dynamic Resource Allocation in the Internet of Vehicles," in IEEE Transactions on Industrial Informatics, vol. 17, no. 7, pp. 4957-4967, July 2021, doi: 10.1109/TII.2020.3019386.

[12] Y. Chen, Z. Liu, Y. Zhang, Y. Wu, X. Chen and L. Zhao, "Deep Reinforcement Learning-Based Dynamic Resource Management for Mobile Edge Computing in Industrial Internet of Things," in IEEE Transactions on Industrial Informatics, vol. 17, no. 7, pp. 4925-4934, July 2021, doi: 10.1109/TII.2020.3028963.

[13] W. Wu, P. Yang, W. Zhang, C. Zhou and X. Shen, "Accuracy-Guaranteed Collaborative DNN Inference in Industrial IoT via Deep Reinforcement Learning," in IEEE Transactions on Industrial Informatics, vol. 17, no. 7, pp. 4988-4998, July 2021, doi: 10.1109/TII.2020.3017573.

[14] P. Zhang, C. Wang, C. Jiang and Z. Han, "Deep Reinforcement Learning Assisted Federated Learning Algorithm for Data Management of IIoT," in IEEE Transactions on Industrial Informatics, vol. 17, no. 12, pp. 8475-8484, Dec. 2021, doi: 10.1109/TII.2021.3064351.

[15] Hussein, A. ., & Mohamed , R. . (2023). Feasibility Study for Developing an Event Prioritizing System Using CMSs. International Journal of Intelligent Systems and Applications in Engineering, 11(1s), 30–45. Retrieved from https://ijisae.org/index.php/IJISAE/article/view/2474

[16] G. Manogaran, J. Ngangmeni, J. Stewart, D. B. Rawat and T. N. Nguyen, "Deep-Learning-Based Concurrent Resource Allocation Method for Improving the Service Response of 6G Network-in-Box Users in UAV," in IEEE Internet of Things Journal, vol. 10, no. 4, pp. 3130-3137, 15 Feb.15, 2023, doi: 10.1109/JIOT.2021.3119336.

[17] M. Liu, F. R. Yu, Y. Teng, V. C. M. Leung and M. Song, "Performance Optimization for Blockchain-Enabled Industrial Internet of Things (IIoT) Systems: A Deep Reinforcement Learning Approach," in IEEE Transactions on Industrial Informatics, vol. 15, no. 6, pp. 3559-3570, June 2019, doi: 10.1109/TII.2019.2897805.

[18] W. Zhang et al., "Deep Reinforcement Learning Based Resource Management for DNN Inference in Industrial IoT," in IEEE Transactions on Vehicular Technology, vol. 70, no. 8, pp. 7605-7618, Aug. 2021, doi: 10.1109/TVT.2021.3068255.

[19] Z. Shi, X. Xie, H. Lu, H. Yang, M. Kadoch and M. Cheriet, "Deep-Reinforcement-Learning-Based Spectrum Resource Management for Industrial Internet of Things," in IEEE Internet of Things Journal, vol. 8, no. 5, pp. 3476-3489, 1 March1, 2021, doi: 10.1109/JIOT.2020.3022861.

[20] Mr. Nikhil Surkar, Ms. Shriya Timande. (2012). Analysis of Analog to Digital Converter for Biomedical Applications. International Journal of New Practices in Management and Engineering, 1(03), 01 - 07. Retrieved from http://ijnpme.org/index.php/IJNPME/article/view/6

[21] W. Fan, S. Li, J. Liu, Y. Su, F. Wu and Y. Liu, "Joint Task Offloading and Resource Allocation for Accuracy-Aware Machine-Learning-Based IIoT Applications," in IEEE Internet of Things Journal, vol. 10, no. 4, pp. 3305-3321, 15 Feb.15, 2023, doi: 10.1109/JIOT.2022.3181990.

[22] S. Tang, L. Chen, K. He, J. Xia, L. Fan and A. Nallanathan, "Computational Intelligence and Deep Learning for Next-Generation Edge-Enabled Industrial IoT," in IEEE Transactions on Network Science and Engineering, 2022, doi: 10.1109/TNSE.2022.3180632.

[23] Davis, W., Wilson, D., López, A., Gonzalez, L., & González, F. Automated Assessment and Feedback Systems in Engineering Education: A Machine Learning Approach. Kuwait Journal of Machine Learning, 1(1). Retrieved from http://kuwaitjournals.com/index.php/kjml/article/view/102

[24] Lokesh Bommisetty, T.G. Venkatesh, Resource Allocation in Time Slotted Channel Hopping (TSCH) Networks Based on Phasic Policy Gradient Reinforcement Learning, Internet of Things, Volume 19, 2022, 100522, ISSN 2542-6605, https://doi.org/10.1016/j.iot.2022.100522.

[25] L. Yang, M. Li, P. Si, R. Yang, E. Sun and Y. Zhang, "Energy-Efficient Resource Allocation for Blockchain-Enabled Industrial Internet of Things With Deep Reinforcement Learning," in IEEE Internet of Things Journal, vol. 8, no. 4, pp. 2318-2329, 15 Feb.15, 2021, doi: 10.1109/JIOT.2020.3030646.

[26] Rosenberger, J., Urlaub, M., Rauterberg, F., Lutz, T., Selig, A., Bühren, M., & Schramm, D. (2022). Deep reinforcement learning multi-agent system for resource allocation in industrial internet of things. Sensors, 22(11), 4099.

[27] Wiling, B. (2021). Locust Genetic Image Processing Classification Model-Based Brain Tumor Classification in MRI Images for Early Diagnosis. Machine Learning Applications in Engineering Education and Management, 1(1), 19–23. Retrieved from http://yashikajournals.com/index.php/mlaeem/article/view/6

[28] Boobalan, P., Ramu, S. P., Pham, Q. V., Dev, K., Pandya, S., Maddikunta, P. K. R., ... & Huynh-The, T. (2022). Fusion of federated learning and industrial Internet of Things: A survey. Computer Networks, 212, 109048.

[29] Tang, S., Pan, Z., Hu, G., Wu, Y., & Li, Y. (2022). Deep Reinforcement Learning-Based Resource Allocation for Satellite Internet of Things with Diverse QoS Guarantee. Sensors, 22(8), 2979.

[30] Li, M., Pei, P., Yu, F. R., Si, P., Li, Y., Sun, E., & Zhang, Y. (2022). Cloud–Edge Collaborative Resource Allocation for Blockchain-Enabled Internet of Things: A Collective Reinforcement Learning Approach. IEEE Internet of Things Journal, 9(22), 23115-23129.

[31] Wang Wei, Natural Language Processing Techniques for Sentiment Analysis in Social Media , Machine Learning Applications Conference Proceedings, Vol 1 2021.

[32] Fang, C., Xu, H., Yang, Y., Hu, Z., Tu, S., Ota, K., ... & Liu, Y. (2022). Deep-reinforcement-learning-based resource allocation for content distribution in fog radio access networks. IEEE Internet of Things Journal, 9(18), 16874-16883.

[33] Jayalaxmi, P. L. S., Kumar, G., Saha, R., Conti, M., Kim, T. H., & Thomas, R. (2022). DeBot: A deep learning-based model for bot detection in industrial internet-of-things. Computers and Electrical Engineering, 102, 108214.

[34] Yang, Z., Yang, R., Yu, F. R., Li, M., Zhang, Y., & Teng, Y. (2022). Sharded blockchain for collaborative computing in the Internet of Things: Combined of dynamic clustering and deep reinforcement learning approach. IEEE Internet of Things Journal, 9(17), 16494-16509.

[35] Gong, Y., Yao, H., Wang, J., Li, M., & Guo, S. (2022). Edge intelligence-driven joint offloading and resource allocation for

future 6G industrial internet of things. IEEE Transactions on Network Science and Engineering.

[36] Olatinwo, S. O., & Joubert, T. H. (2022). Deep learning for resource management in Internet of Things networks: a bibliometric analysis and comprehensive review. Institute of Electrical and Electronics Engineers.

[37] Tian, K., Chai, H., Liu, Y., & Liu, B. (2022). Edge intelligence empowered dynamic offloading and resource management of MEC for smart city Internet of Things. Electronics, 11(6), 879

[38] Lai, X., Hu, Q., Wang, W., Fei, L., & Huang, Y. (2020). Adaptive resource allocation method based on deep Q network for industrial Internet of Things. IEEE Access, 8, 27426-27434.

[39] Liang, Fan, et al. "Toward Deep Q-Network-Based Resource Allocation in Industrial Internet of Things." IEEE Internet of Things Journal 9.12 (2021): 9138-9150.