



ORIGINAL ARTICLE

OPTIMIZING PREDICTIVE MAINTENANCE IN SMART MANUFACTURING USING DEEP REINFORCEMENT LEARNING-DRIVEN INDUSTRIAL AI SYSTEMS

^{1*}Poovarasana P

1 IB service department, VST Tillers Tractors Ltd, SIPCOT Industrial Complex, Hosur, India,

Corresponding Author Email: poovarasana@vsttractors.com

(Received 28 January 2025; revised 27 March 2025; accepted 02 May 2025)

Abstract

Preventative maintenance has become a vital tool in the age of Industry 4.0, helping to maximize machine efficiency and reduce downtime in smart industrial settings. In this research, deep reinforcement learning (DRL) coupled with industrial artificial intelligence (AI) systems is used to propose an innovative method for preventative maintenance. Through the utilization of DRL's capacity to acquire suitable servicing policies via communication with machine information in real-time, the suggested system adjusts itself dynamically to changing operating circumstances. The creation of a multi-agent DRL framework, which reduces unplanned downtime and boosts operational effectiveness by predicting equipment breakdowns and scheduling maintenance tasks in real time, is one of the key achievements. In addition, based on artificial intelligence, analytics are used to improve the algorithm to evaluate machine wellness, allocate resources optimally, and prolong equipment lifespan. The suggested solution outperforms conventional maintenance techniques in terms of precision and flexibility, opening the door for more effective, self-optimizing manufacturing techniques. Experiments conducted in a smart manufacturing environment demonstrate notable gains being manufacturing continuity, decision-making effectiveness, and operational reduction in costs.

Keywords: *predictive maintenance; deep reinforcement learning; industrial artificial intelligence; smart manufacturing; multi-agent systems; machine health monitoring; maintenance optimization; industry 4.0; equipment failure prediction*

1. Introduction

The maintenance process takes up 15 to 40 percent of the total cost of production in the plants. Machines and resources experience different types of failures, which affect the efficiency of systems, exacerbate the well-being of machines, and eventually cause their eventual failure. Unless machines are switched off, they will be worn out, which will require servicing. Maintenance according to [1] refers to a set of processes employed to restore an object to a state at which it can perform its intended functions. The benefits of reducing unexpected downtime of machines are lower service and manufacturing costs, greater machine reliability, and a small number of machine breakdowns. Similar to most planning problems, a continued scheduling and preparation problem is also an optimization problem to design effective repair policy and properly allocate restoration tasks and resources to the geographic problem. The enhancement of maintenance strategy improvement in literature is generally defined as a multi parameter enhancement whose objective is to find out the appropriate upkeep actions and the time of these actions in dynamic environment to ensure high yield, cost efficient maintenance and accessibility of devices using a set of constrained or unconstrained preservation capabilities [2]. The maintenance of the machine/assets as mentioned above is inevitable and that is why proper maintenance scheduling is essential in ensuring that assets are available as much as possible at the minimal cost. The short- or long-term costs and effects of not well thought of manufacturing and service processes are beyond the capacity of most companies as these might result in failure to meet consumer demand and the company losing sales. This renders many industries to be unable to remain competitive. The expenses of planned and unforeseen equipment downtimes and maintenance expenditure can be reduced through the

development of proper maintenance plans. The two overall categories into which maintenance approaches belong are Corrective Maintenance (CM) and preventive maintenance (PM) [3].

The maintenance management and optimization problem helps to solve the optimal maintenance practices under a chosen maintenance technique. Maintenance of breakdowns, also called corrective maintenance, is a reactive approach, which implements the run-to-failures approach. Maintenance is only done after faults have happened to the machine. In the event of equipment failure through this method, a different or major overhaul is usually required to resume the machine with operating conditions and maintenance repair costs and time spent to do so is usually very high [4]. One of the major drawbacks of the preventative care approach is that it does not give decision-makers an opportunity to strategize on how to continue with operations, beyond the costs incurred in doing so. Machine breakages are unpredictable and they interfere with production or the running of operations. This is not only possible to cause huge losses due to non-planned down time but also requires costly and time-consuming repairs. It may also lead to incidences; less important equipment tends to embrace the preventative care method. Conversely, preventative maintenance aims at enhancing machine reliability through proactive preventive maintenance affecting machines against mechanical breakdowns [5]. This will reduce chances of breakdowns. Coming up with preventative maintenance measures that would ensure smooth and efficient productivity is a tricky task, however, because PM measures may cause damage to the production system and lead to additional losses in production. The cost of a large-scale PM may overshadow its benefits whereas insufficient PM can also be useless in avoiding unwanted cases of equipment failures. The nonlinear system dynamics that emerge as a result of the machinery conversation makes it hard to balance the delicate decision trade-offs which are made in PM with industrial systems [6].

There are two main types of preventative maintenance strategies, namely, condition-based and scheduled service. Scheduled maintenance is a form of preventative maintenance whereby some maintenance activities have been set and performed at a set time interval and it is performed in the presence or absence of any apparent sign of wear and tear or malfunctions. This strategy may be rather violent and eventually create unnecessary expenditures, such as sustenance of the cost of resources and the prices of spare parts of equipment or facilities that are still in good state, despite the fact that it leads to fewer unexpected failures [7]. The expenses of planned maintenance, production, and the industry-related costs of the intervention will be very expensive in case maintenance plans are not optimized. Lastly, planned maintenance cannot prevent any undesirable surprises because of unpredictable, changing, and nonlinear system behavior that is caused by machine conversations, and that may cause machinery to malfunction before the specified maintenance time is met [8].

More frequently referred to as maintenance based on conditions, predictive maintenance suggests the implementation of maintenance procedures through evaluation of the entire system or component in advance of failure. Due to its high effectiveness and low costs compared to other methods, preventative care has been identified as one of the more appealing management contributions to the manufacture of equipment [9]. Such a strategy is used to prevent unnecessary maintenance spending created in the planned maintenance approach and significantly decrease the unscheduled failures because the repair decisions are made on the basis of transforming, real-time equipment medical states. Numerous studies have been conducted in the scientific community to leverage the prediction model of machine-learning based approaches that can be used to prognosticate, diagnose, and accurately predict machine faults. Another effort at enhancing the prediction results of machine learning-based models of prediction, which rely on large amounts of information, is using generative models to generate larger datasets. In a case in point, [10] used some flawed samples and conditional generative adversarial network to generate flawed information using regular information. The rationale of this effort was to eliminate the issue of unbalanced datasets that engineers face in an attempt to implement data-driven forecasting methods. Nowadays, companies must cope with mass customization and the speedy pace of development, and it poses a serious challenge to intelligent production sites. They must have the capabilities of operating under very unpredictable market conditions and fulfilling the continually more demanding criteria of sustainable development and product quality within the shortest period of time [11].

To address these issues, Germany initiated Industry 4.0 in 2013 to aid in the construction of flexible and adaptive manufacturing systems. The current use of Industry 4.0 implementations cannot use similar

advanced means such as machine learning, yet the potential of the initiative is enormous and can potentially impact. In contrast to modeling, virtualization, or big data methods, which are increasingly taking center-stage in the scope of the study of production, machine learning does not. This opinion has been disproved earlier because it emphasized the broad applicability of machine learning in modern manufacturing and their ability to get exceptional results [12]. To elaborate more, our discussion focuses on the benefits of deep Reinforcement Learning (RL) to production systems and RL as a data-driven online optimization method is also considered. The 3 primary subfields of machine learning are reinforcement learning, unsupervised learning, and semi-supervised learning. Although both supervised and unsupervised learning require a previously labeled set of data, the difference between the two is that reinforcement learning entails learning through direct interaction with its immediate environment [13]. It does not need any information collected beforehand or previous (human) knowledge, but learns through its experimentation and is flexible and able to adjust to new situations. Considering these flexible and desirable attributes in modern-day manufacturing, this study tries to record the state of the art in real or virtual deployment of deep reinforcement learning in production systems. Existing problems lead to the establishment of new research fields [14]. One area of machine learning, RL, is distinguished both by the lack of unsupervised learning and the lack of supervised learning, that is, RL uses experimentation to learn in close connection with its environment. It is used whenever they require a real-time, reaction-based decision-making process in dynamic situations; it does not require any monitoring or defined labeled or unstructured set of data. Transform by extrapolating the already learnt information and then make sequential choices in acclimating to the altering conditions in the environment online [15].

2. Related Works

The amount of literature that has been published on problems of integrated production and maintenance has grown exponentially over the past years. The difference between publications in this classification is significant in a number of aspects, including the way the optimization problem is outlined, the number of approaches to the problem that are employed in the solution, and the outline and core values of the system that is being studied. Single-machine systems are discussed but flow lines, which are systems consisting of a large number of devices that are connected in series and separated by buffer [16]. In that, it is a combination of a production plant and an output buffer and another storage location, investigates a comparatively distinct system. They must be transported to the output buffer storage where they are then transferred to the inventory where they are then transferred to the clients. The manufacturing systems which are considered by the scientific literature to date are either known to produce one type of products or multiple types of products. The reason behind assuming the decision-maker will always be aware of the manufacturing facility in degraded condition is because of the decision-maker [17]. In some of the publications the deteriorating condition is determined through periodic evaluations, but both incomplete and periodic examinations. The dominant assumption of the majority of published literature is that a faulty system can continue to produce satisfactory end-items until it fails, at typically a lower service rate. Poor manufacturing standards have been considered, where an object of the created system causes a part of the objects to be nonconforming with regards to quality [18]. Both large and small maintenance operations are possible. Although, minor maintenance only reduces the extent of degradation of the production system, big maintenance restores the production system to its original state. Maintenance activities are typically linked with time or monetary costs of the activity; the acquisition of the necessary replacement items is also a considered aspect. The available literature on combined manufacturing and upholding control can be categorized according to the decision-making process that is related to an infinite or finite horizon [19]. Costs in manufacturing, supplies, back order, and ongoing costs are normally considered when developing the optimum manufacturing/maintenance strategy or strategy. The models and the solution methodologies also differ greatly, and include approaches such as dynamical programming and reinforcement learning methodologies to mixed-integer and linear-quadratic methods of computing. Periodic evaluations are used to gauge the degeneration of the system whereas threshold-based policies are used to guide management choices [20]. There are two related mixed integer optimization problems that are solved by an extended Varangian evolutionary approach and generating mathematical formulas of numerous performance measures. The time-dependent failures of a discrete-time manufacturing process, preventive/corrective regular consumption and continuous demand are

considered. A decision-making model with Markov process of decision making is employed to develop joint manufacturing and upkeep regulations and the structure of the optimal policy is looked into. An stochastic process of continuous Markovian failure of machines is investigated which includes disappointment, assessment, regular consumption and repair time. Twofold threshold is a policy that is used when making decisions about minor and major maintenance. Mean-time-to-fail measurements and availability have the mathematical formulations which are closed. There is an unreliable stochastic production / inventory system that utilizes the (s, S) strategy. It develops the corresponding Markov chain framework and develops equations of the measures of the system efficiency, median stock level, and service level. They are on a mission to find out the most appropriate maintenance plan of an objective activity that involves savings of service and maintenance costs per unit time, and an increased income through the increased levels of service. It analyzes a stochastic single-machine system, which passes through various stages of degradation. Postponement as well as lost sales is acceptable. The issue of formulating joint rules of production maintenance management is represented by a semi-Markov approach to decision-making. Dynamic programming is used to obtain the optimal course of action to minimize the cost of inventory, which involves restoration as well as lost sales costs.

3. Artificial Intelligence Facilitates the Development of Intelligent Manufacturing

The entire system and lifecycle of product creation is all one brought together through the use of the technological tools of intelligent manufacturing, a new manufacturing paradigm, which is integrated with novel technologies of communications and information, knowledgeable technological advances, large manufacturing technologies (including design, manufacture, leadership, examination, and connection) and related technologies of products. The production life cycle, therefore, utilizes self-contained detecting, connectivity, interaction, education, assessment, mental processes, decision making, oversight and the execution of human beings, device, substance, and ecological data to make several elements of a manufacturing business/category work properly and efficiently. This streamlines production and yields quality and affordable services to customers that are environmentally friendly. Through this, a manufacturing group or organization is able to be more competitive in the market. The AI technology in the area of smart manufacturing simplifies the development of new models, methods, and forms along with structure of the system and technological systems.

New models: proactive manufacturing framework that is sociable, flexible, adjustable, serviceable and internet connected that assists with production and customer support. The new means involve digitization, Internet of Things, utilization of virtualization assistance, collaboration, personalization, flexibility, and thinking in intelligent production processes between humans and machines. The new form is an intelligent and productive ecosystem with mass innovation, creativity, cross-border integration, independent information and with pervasive connectedness. The profound integration of the implementation of different designs, procedures, and shapes presented in Figure 1 will eventually create a network of smart manufacturing.

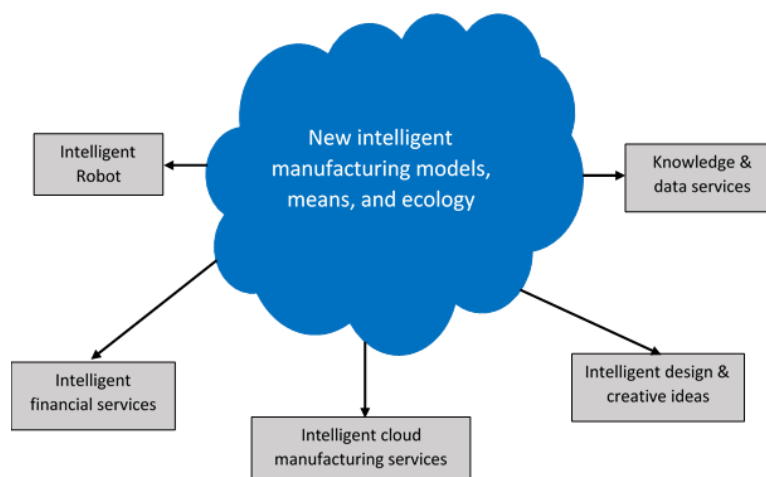


Figure 1: Components of Intelligent Manufacturing

3.1 Deep Reinforcement Learning

It is an RL that is based on fixed optimization equations, but RL methods are in essence sequential and responsive. Due to this fact, most experimental methods become exposed to optimality barriers; they are the curse of dimensionality and the punishment of history, both of which become especially relevant in problems with continuous state space, field sensitivity, and long-range selection. Deep reinforcement learning methods are used when tabular methods are infeasible to estimate the values in the current state with the help of function approximates. The deep reinforcement learning method is based on using a neural network to compute the different states rather than having to map each state to its quantities in a tabulated form. It simplifies the solution space of the decision-making processes.

3.2 The RL or DRL solution and maintenance planning problem

Within this section, advantages of reinforcement learning as opposed to other optimization schemes such as heuristics and meta-heuristics, as well as why it is a suitable solution to the current management problem will be discussed.

1. The nature of the industrial or manufacturing environment makes the ongoing planning issue a pretty hard optimization problem given that it is dynamic and stochastic in nature. The sequence of events or states is dynamic; rather, the complex, non-linear relationships among the surrounding organizations, including the situation of the industry/factory operations, the limited supply influences, structural factors among other units within the framework, etc., can influence the unit(s)/system rate of degradation and websites. And thus they act in a random way. Since the techniques utilized by RL rely on the Markov Decision Process (MDP) structure, which is stochastic, RL is able to learn dynamic preservation regulations and not merely static ones because these approaches are applicable to planning in stochastic environments (MDPs can be applied to planning in stochastic environments).

2. Second, the RL agents can be interested in maximizing complex sequential decisions when there is uncertainty since it allows the agent to interact with the environment when making decisions. Provisional settlements about maintenance effected in the environment of a maintenance planning problem may initially benefit the maintenance costs only to lead to the reduction of the cost of factory operations. This is one such instance of delayed reward situation in which the decision made may not only affect the immediate gain but also the situations that arise out of the decision and hence all the actions that arise out of it. The delayed-reward functionality is one of the peculiar features of reinforcement learning besides the search functionality through trial and error. The fact that RL can find solutions on the first hand, based on the interactive goal-seeking behavior, contrasts with other alternatives, which only aim to solve issues once they know how they fit in the real world or immediate decision-making. It is another crucial characteristic of RL that causes the need to carry out the planned maintenance issue.

3. To be more specific, the route of machine deterioration, industry/production ecosystem model, and the associations within the several entities could be rather difficult to quantify in relation to the condition-based maintenance techniques. RL can be used together with the model-free technique that can assist in managing non-linear complex systems. The deterioration of the performance quantities or the amount of life that the machine can perform can be used as a symbol of the machine's health conditions. The approach of showing the way of destruction or checking the quantity of useful life which is alive is not that easy; it requires an in-depth knowledge of preservation data analysis. Assuming that the health conditions are not represented in a precise form and are not a true reflection of the actual machine condition, this error replicates and influences the decision in spite of the maximization of efficiency method which has been employed. Condition-based upkeep policies utilize this data to directly project the machine health conditions into upkeep responses. With RL to exploit the efficacy of methods to deep learning we can resist the flaw prognosis or decay exhibiting model and instantly project contests to the optimal actions by training on the situation. This is the other benefit of RL compared to other methods of maximizing efficiency. The agent will decide when the servicing will be done and the likelihood of the machine breaking down at every stage.

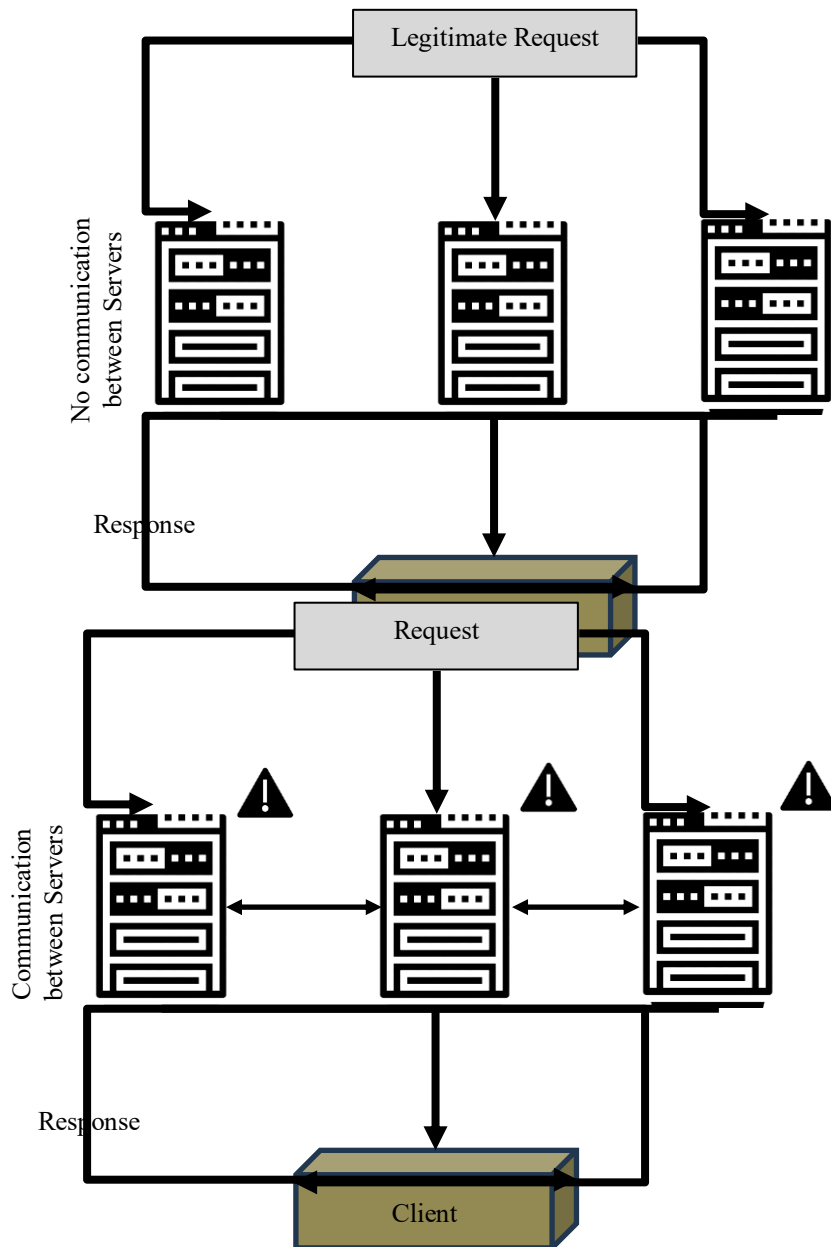


Figure 4: Architectural Diagram for Proposed Setup, (a) Before the threshold (b) After the threshold

4. The trick in solving an optimization problem lies in the transformation of the optimization objective into the cost and constraint functions mathematically, which is not an easy task. Poor formulation of issues will result in poor policies, which, in turn, might have undesirable outcomes when enacted. However, with the model-free reinforcement learning method, we can learn directly in the environment.

5. And finally, since the RL agent has spent time learning what is the best maintenance action to take in each scenario it has researched should he have been trained right, it will certainly be much faster and realize better results when it comes to inference than heuristics and meta-heuristics.

3.3 Cache Attacks and IoT

The non-standard expansion of the IoT industry has been the primary cause of DDoS attacks. The situation that happened after the introduction of IoT devices has been a race to make everything smarter. IoT can be found in almost all fields; whether it is cities, healthcare, agriculture, and industry. These devices generate and store data of various sizes, depending on the purpose, on scales as small as tiny and as large as monumental. Future businesses have unlimited opportunities in regards to the applications of Industrial IoT (IIoT), which processes data of hundreds of connected sensors. IIoT has a number of advantages. A higher productivity rate, better accuracy, scalability of the network, predictive maintenance, less downtime of the machine, less power use, better safety, and optimizing. Multiple IoT

(MIoT) has numerous applications; one of them is smart cities. The above IoT applications are susceptible to cache attacks as they use the cache to reduce the latency and increase the throughput.

In the absence of any suspicious activity, the situation depicted in Figure 4(a) is a common one. In this case, Memcached servers work as originally planned where there is no intercommunication between them. Based on the suggested design, servers will issue any message when a specific number of requests is obtained with the help of which Memcached will remain functional as desired under the majority of circumstances. The Memcached servers share data when they suspect an attack as in Figure 4(b). The suspected victim i.e. the Memcached server where the individual threshold value has been exceeded sends SYN requests to the rest of the Memcached servers to determine whether or not the attack has occurred. Thresholds are replicated across servers; therefore, it can be assumed that τ is the same across all Memcached servers. Due to this, in the situation of DDoS attacks, even when the latency is affected in any instance, latency is minimally affected. Context-aware computing is also suggested as a further way of lowering performance besides the revised Memcached server. The optimal threshold value varies extensively based on the capabilities of the service supplier; thus, the provider of services should decide it through the context-sensitive computing. IP spoofing is another method of requesting a DDoS attack by malicious actors to overwhelm the victim system. Figure 5 gives an algorithmic representation of a step-by-step representation. The proposed remedy is that an IDS raises alarm whenever there is incursion. Because detection of abnormal activity is the point of paradigm in anomaly detection, the focus in IDS remains on attaining a higher recall rate. The proposed system will be capable of raising the alarm so that the service provider can intervene. Action needs the involvement of human factors; hence the proposed technique works equally well with the Intrusion Prevention System (IPS) without compromising on the performance of the system. This technique is effective since it triggers off an alarm in case even a single server is being compromised and the chances of the attack being mitigated due to an early warning are enhanced.

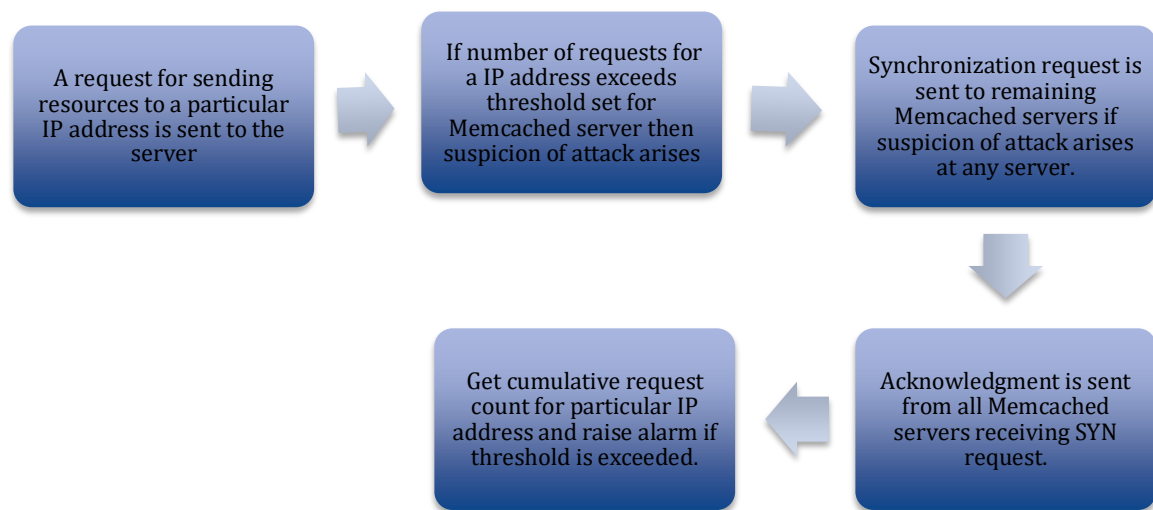


Figure 5: A Detailed Process Flow of the Proposed Solution for DDoS Attack using Memcached Servers

Step 1: A request for sending resources to a particular IP address is sent to the server, and the record is kept as $n(IP_x, \tau)$, where, n is the number of requests, IP_x is the IP address for the i th client, t is time.

Step 2: For any Memcached server m_k , $k \in (1, N)$, where N is the total number of Memcached servers, suspicion of attack arises if for a particular IP address number of requests received at time τ is greater than the threshold value set for that Memcached server, i.e., $n(IP_x, \tau) \geq n_{th}(IP_x, \tau)$ (1)

where, n_{th} Threshold for the number of requests

Step 3: If the conditions $n(IP_x, \tau) \geq n_{th}(IP_x, \tau)$ holds good, then a Synchronization request (SYN) is sent to nearby/all Memcached servers, $SYN(IP_x)$ (2)

Step 4: All Memcached servers receiving SYN request then send the acknowledgment giving information about the number of requests for the suspected victim IP address for the said time duration, i.e., $ACK(n(IP_x, \tau))$ (3)

This scenario is depicted in Figure 6b, where communication between Memcached servers has been represented. All the variables, i.e., n_t , n_{at} , τ are arbitrary and can be updated according to requirements, thus enabling context-aware computing.

Step 5: The cumulative request count is calculated as the sum of requests sent to the Memcached server n_k for the IP address of x-th client and acknowledgment of requests received by other servers for the same client.

$$\{(n(IP_x, \tau))_{m_k} + \sum_x ACK(n(IP_x, \tau))\} \quad (4)$$

the system is probable to be under attack if,

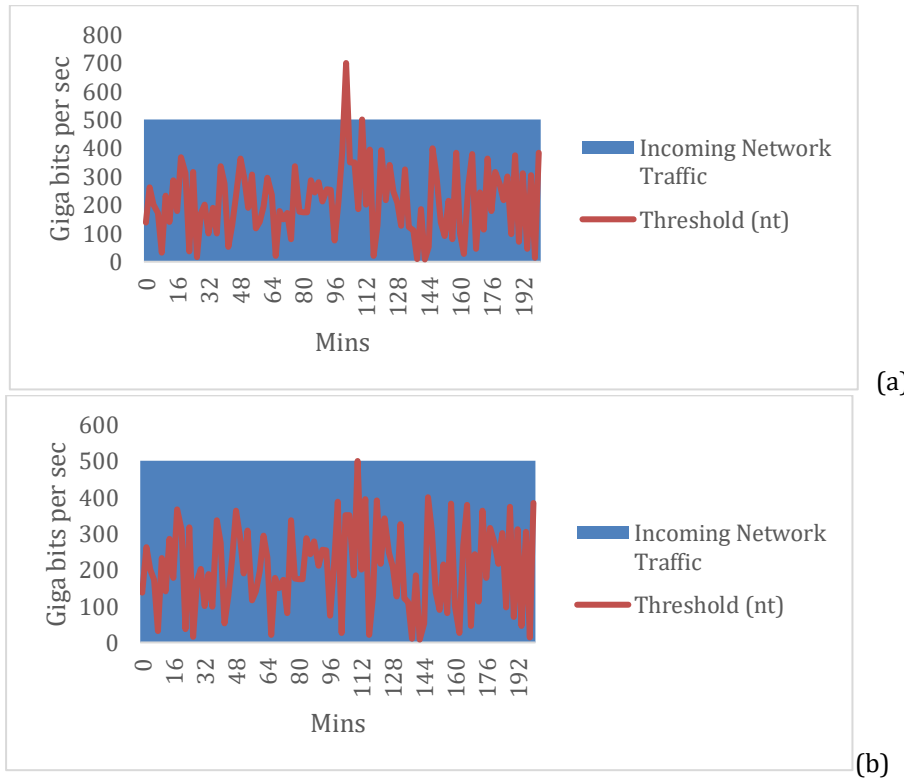
$$\{(n(IP_x, \tau))_{m_k} + \sum_x ACK(n(IP_x, \tau))\} \geq n_{at}(IP_x, \tau) \quad (5)$$

where, n_{at} - Cumulative threshold for the number of requests.

In this instance, research is necessary to eliminate the possibility of assault. Should any Memcached server issue any SYN request over any length of time τ and $(\tau_1 < \tau < \tau_2)$ holds then no additional SYN request should be issued by any other Memcached servers. The cumulative threshold is specified over the entire period $(\tau_1 < \tau < \tau_2)$; each spike that indicates a threat to any Memcached server in the entire time will be counted automatically. Spike may be relaying SYN requests to Memcached servers after τ seconds. The transmission of a single SYN message with a time interval of τ maintains the low latency and high throughput as well as avoiding the deterioration of efficiency

3.4 Case study for detecting DDoS attack using Memcached server

The practical setup is set up on a Linux operative server that has an Intel Core i7 dual core processor, 16GB memory and operating system that is 64bit. A virtual machine is used to record information on datasets. The proposed approach is evaluated using three Memcached servers, as well as a realistic dataset that is representative of the real world. Considering that the objective of Memcached is to optimize dynamic web application, context-aware techniques are applied to devise threshold values which will enable maximum service to be delivered without compromising server performance. The effectiveness of the offered strategy is self-evident in the cases when it can be demonstrated that with the correct choice of threshold, proactive measures can be taken by increasing the alarm in case of a DDoS attack.



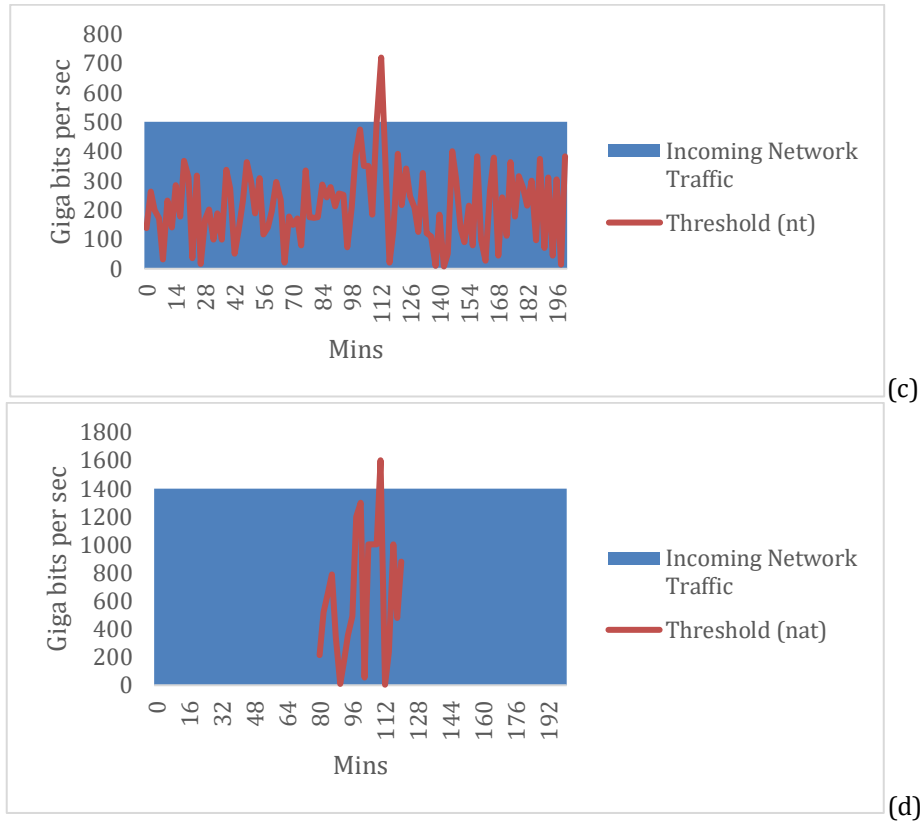


Figure 6: Case study of Proposed Solution for (a) Memcached server 1 with a threshold value n_t (b) Memcached server 2 with threshold value n_t (c) Memcached server 3 with threshold value n_t and (d) cumulative result for raising the alarm with a cumulative threshold value n_{at}

Figure 6(a) indicates that at point a1, Memcached server 1 has a threshold, n_t , of more than 100 minutes; at this point, the SYN command is sent to all Memcached servers to be acknowledged. Figure 6(b) indicates that there is no surge on Memcached server 2. Memcached server 3 has a peak at minute 110, or at point f1 in Figure 6(c) although Memcached server 1 does not surpass the threshold at minute 100 when it transmits a SYN instruction. That is to say that invasion does not have to take place at the same time to all the servers. Figure 6(d) contains cumulative results of all Memcached servers between the period ($\tau_1 < \tau < \tau_2$). No recognition will be provided because there are no SYN requests at time earlier than ($80 < \tau < 120$) because Figure 6(d) indicates that nothing is provided at that time period. Also the alarm is activated at point g1 at minute 110 as opposed to minute 100 when the SYN request was initially sent. This implies that the type of approach proposed will avoid instances of false positives and will also avoid delay as SYN and ACK requests are required once even in cases where the duration of time has passed. Figure 6(d) contains cumulative results of all Memcached servers between the period ($\tau_1 < \tau < \tau_2$). No recognition will be provided because there are no SYN requests at time earlier than ($80 < \tau < 120$) because Figure 6(d) indicates that nothing is provided at that time period. Also the alarm is activated at point g1 at minute 110 as opposed to minute 100 when the SYN request was initially sent. This implies that the type of approach proposed will avoid instances of false positives and will also avoid delay as SYN and ACK requests are required once even in cases where the duration of time has passed.

3.5 Federated Data Creation

In order to build federated datasets, the datasets are converted to binary labeled information and divided into segments. The volume of datasets is kept constant, however, in case when the original dataset is reshaped to binary distribution, smaller information fragments contain different kinds and distributions of assaults which can be viewed as real-world events. There may be different data distributions such as in real world implementation; this has been taken into account by shuffling with extreme data generation to generate divided data sets. Based on these two datasets, two segmented datasets that represent many edge devices are obtained. Figure 7 shows the mean pattern of information distribution of thirty randomly selected sections of each data set.

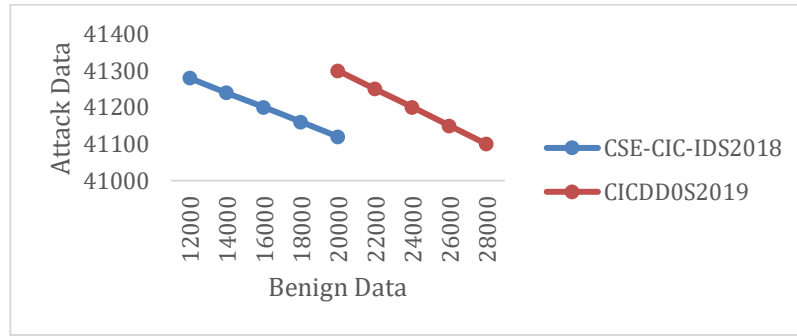


Figure 7: Attack and Benign Data Distribution of CSE-CIC-IDS2018 and CICDDoS2019 dataset

According to Figure 7, there is a close distribution of attack and benign data in the CSE-CIC-IDS2018 Dataset since in every category of information, the benign information is between 42.1 and 42.9 whereas the information about the attack is between nearly 57.1 and 57.9. Conversely, the CICDDoS2019 data records an attack and benign data of 76.1% and 23.9 percent, respectively, in each of the subsets. The computer model will be provided with different information distributions in the real world, therefore this variation in data distribution is to imitate that scenario.

4. Results and Discussions

The models are evaluated based on the effectiveness of the models in a similar situation, including batch sizes, fixed epochs, communication rounds, and the same hyper parameters. Timing of execution is an issue that is to be considered when it comes to measuring the quality of any IDS approach. Execution time of each model is compared with a single epoch, the results are satisfactory almost with all models, and they are also compared with different epochs to investigate the trend of execution time and performance of a single communication round. One major problem with edge devices is that they are of small sizes; in order to address the problem, the original and updated model sizes are compared. To more closely consider four examples are provided below.

4.1 Model performance for Varying Communication Rounds fixed Epochs, and fixed Batch Size.

Four communication rounds are considered and these include 2, 5, 8 and 15. The curve of accuracy and time of each of the four deep learning models and four interaction cycles is illustrated in figure 8. The results on the CSE-CIC-IDS2018 dataset.

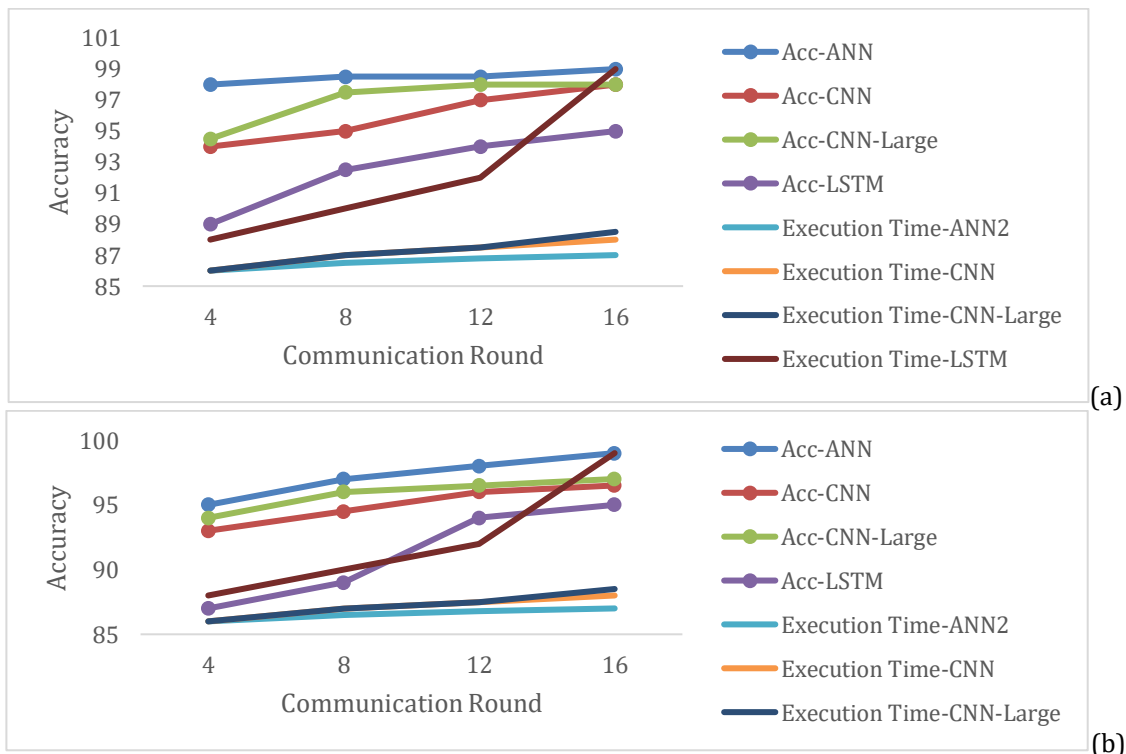


Figure 8: Comparison of Accuracy and execution time of Models for varying Communication Rounds, fixed Epochs, and Batch Size for (a) CSE-CIC-IDS2018, (b) CICDDoS2019 Dataset

Figure 8(a) indicates that the accuracy of ANN and the time required to complete each round of communication are best compared to those of other methods. Regarding accuracy, CNN-Large will rank secondly, as the highest accuracy of 98.9 is achieved at 15 communication rounds, however the communication rounds make the time soar rapidly. Given that LSTM takes the longest time to execute and the reliability in LSTM cannot be equated with other models, its efficiency may be enhanced in comparison to other models. A maximum of 96.56 percent accuracy can be achieved at 15 communication rounds by LSTM. There is no significant difference in the execution time of CNN and ANN but ANN is much better with regards to the accuracy that it achieves in a given duration. The results based on the CICDDoS2019 data are presented in Figure 8(b). Models will again take a similar direction, ANN will be the most efficient with the quickest in-speed with 99.2 accuracy. On the other hand, LSTM is the poorest in terms of implementation time and precision. The highest accuracy that LSTM can reach at 50 epochs is 96%. The results of Figure 9 indicate the accuracy and time curve of all four deep learning systems and four interaction cycles. All the models except LSTM possess precisions more than 0.9 in all rounds of communication and data sets. Accuracy on LSTM is also achieved with precision more than or equal to 0.9 following five rounds of communication. ANN is best in all communication cycles in terms of precision as shown in Figure 9.

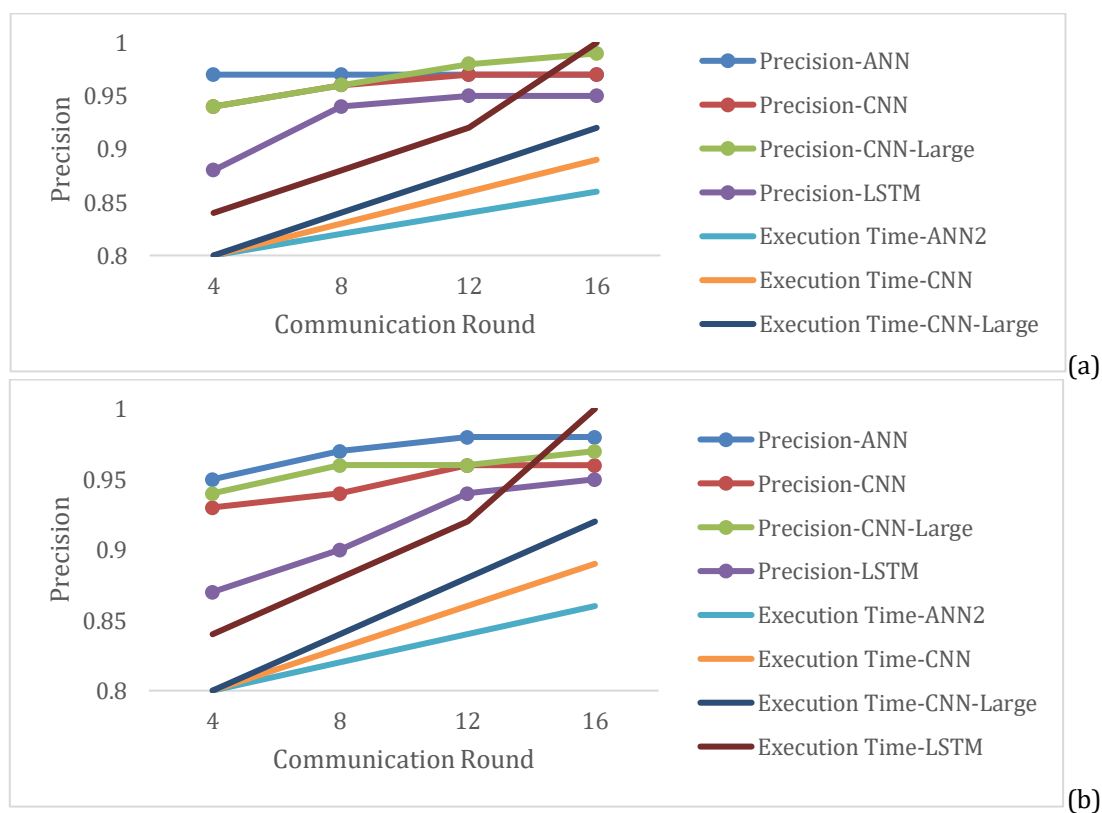
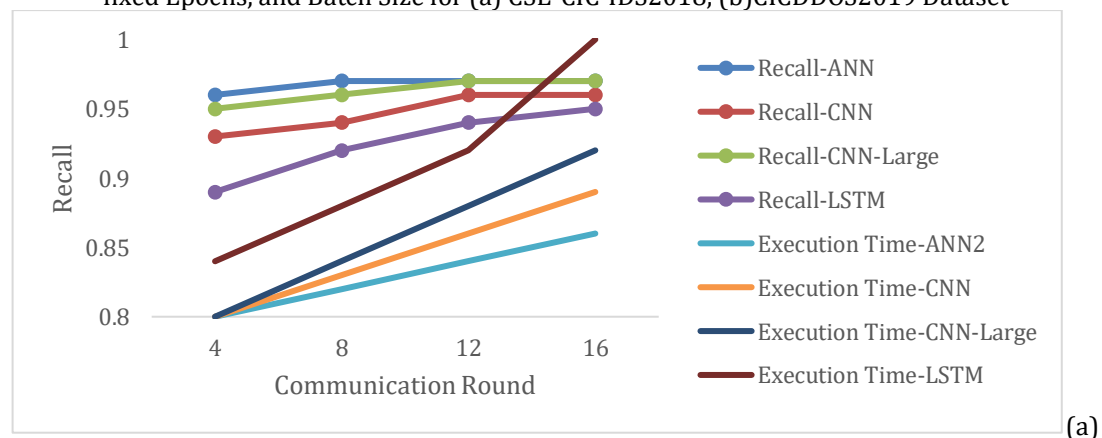


Figure 9: Comparison of Precision and execution time of Models for varying Communication Rounds, fixed Epochs, and Batch Size for (a) CSE-CIC-IDS2018, (b) CICDDOS2019 Dataset



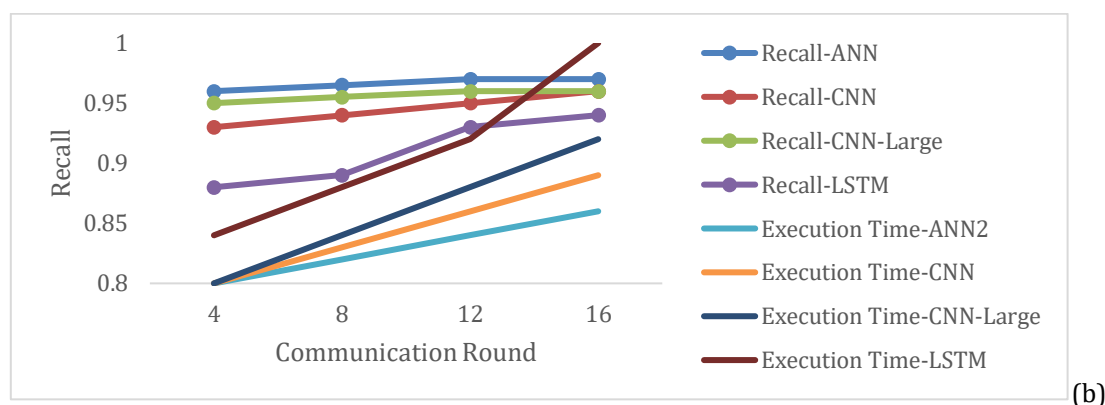


Figure 10: Comparison of Recall and execution time of Models for varying Communication Rounds, fixed Epochs, and Batch Size for (a) CSE-CIC-IDS2018, (b) ICDDOS2019 Dataset

In the case of CSECIC-IDS2018, an LSTM required 5 rounds of communication to get a value to recall higher than 0.9, as opposed to CICDDoS2019 which required 8 rounds of communication. Figure 10 demonstrates that ANN is the most successful in recall during the entire communication rounds. Figure 10 illustrates the recall rate and the time curve of each of the four deep learning models and four communication cycles. All models have a recall of over 0.9 except in the cases of LSTM with both datasets and all communications rounds.

5. Conclusion

In conclusion, a hybrid deep learning and edge computing system that enables the implementation of AI and IoT into the VANETs is a feasible solution to the issues of modern smart public transportation systems. Using this approach, Vehicle-to-Everything (V2X) communication is significantly improved, and the latency is reduced to a minimum to allow processing and making decisions regarding the information in real-time. With enhanced traffic management, safety, and coordination of autonomous cars, AI-based approaches would provide a credible, scalable urban network infrastructure. According to the results of this work, significant improvements in communication quality, network reliability, and overall performance of the system in general, the potential to transform VANETs and open the door to more intelligent and healthier transportation networks can be noted, which is supported by AI and IoT.

Conflict of Interest Statement

There is no conflict of interest

Data Availability Statement

Data restricted but available based on reviewer's request

Authors' contributions

A: Methodology, Writing, Implementation. Validation

Funding

No funding

References

- [1].Shinde, R. K., et al. "Intelligent IoT (IIoT) Device to Identifying Suspected COVID-19 Infections Using Sensor Fusion Algorithm and Real-Time Mask Detection Based on the Enhanced MobileNetV2 Model." *Healthcare*, vol. 10, no. 3, MDPI, Feb. 2022, p. 454.
- [2].Lin, H., et al. "Toward Secure Data Fusion in Industrial IoT Using Transfer Learning." *IEEE Transactions on Industrial Informatics*, vol. 17, no. 10, 2020, pp. 7114–7122.
- [3].Monica, M., et al. "PMSG Based WECS: Control Techniques, MPPT Methods and Control Strategies for Standalone Battery Integrated System." *AIP Conference Proceedings*, vol. 2405, no. 1, AIP Publishing LLC, Apr. 2022, p. 040013.
- [4].Saha, D., et al. "Application of Nanotechnology in Neural Growth Support System." *2022 IEEE 2nd Mysore Sub Section International Conference (MysuruCon)*, IEEE, Oct. 2022, pp. 1–6.
- [5].Raviprasad, B., et al. "Accuracy Determination Using Deep Learning Technique in Cloud-Based IoT Sensor Environment." *Measurement: Sensors*, vol. 24, 2022, p. 100459.

- [6].Sun, X., and K. Shu. "Application Research of Perception Data Fusion System of Agricultural Product Supply Chain Based on the Internet of Things." *EURASIP Journal on Wireless Communications and Networking*, 2021, pp. 1–18.
- [7].Shinde, R. K., et al. "Intelligent IoT (IIoT) Device to Identifying Suspected COVID-19 Infections Using Sensor Fusion Algorithm and Real-Time Mask Detection Based on the Enhanced MobileNetV2 Model." *Healthcare*, vol. 10, no. 3, MDPI, Feb. 2022, p. 454.
- [8].Munir, A., et al. "Artificial Intelligence and Data Fusion at the Edge." *IEEE Aerospace and Electronic Systems Magazine*, vol. 36, no. 7, 2021, pp. 62–78.
- [9].Yang, F., et al. "Internet-of-Things-Enabled Data Fusion Method for Sleep Healthcare Applications." *IEEE Internet of Things Journal*, vol. 8, no. 21, 2021, pp. 15892–15905.
- [10]. Garikapati, P., et al. "K-means Partitioning Approach to Predict the Error Observations in Small Datasets." *International Journal of Computer Aided Engineering and Technology*, vol. 17, no. 4, 2022, pp. 412–430.
- [11]. Patil, R. R., and S. Kumar. "Rice-Fusion: A Multimodality Data Fusion Framework for Rice Disease Diagnosis." *IEEE Access*, vol. 10, 2022, pp. 5207–5222.
- [12]. Ramaraj, E., and D. N. V. S. L. S. Indira. "Data Fusion Method and Internet of Things (IoT) for Smart City Application." *2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV)*, IEEE, Feb. 2021, pp. 284–289.
- [13]. Chelliah, B. J., et al. "Analysis of Demand Forecasting of Agriculture Using Machine Learning Algorithm." *Environment, Development and Sustainability*, 2022, pp. 1–17.
- [14]. Xu, S., and S. Kunliang. "Application Research of Perception Data Fusion System of Agricultural Product Supply Chain Based on the Internet of Things." *EURASIP Journal on Wireless Communications and Networking*, 2021, pp. 1–18.
- [15]. Mohaghegh, P., et al. "Depth Camera and Electromagnetic Field Localization System for IoT Application: High-Level, Lightweight Data Fusion." *2021 2nd Asia Service Sciences and Software Engineering Conference*, IEEE, Feb. 2021, pp. 94–101.
- [16]. Kliestik, T., et al. "Remote Sensing Data Fusion Techniques, Autonomous Vehicle Driving Perception Algorithms, and Mobility Simulation Tools in Smart Transportation Systems." *Contemporary Readings in Law and Social Justice*, vol. 14, no. 1, 2022, pp. 137–152.
- [17]. Sharma, P. K., et al. "IoT Solutions with Multi-Sensor Fusion and Signal-Image Encoding for Secure Data Transfer and Decision-Making." *arXiv preprint arXiv:2106.01497*, 2021.
- [18]. Durana, P., et al. "Digital Twin Modeling, Multi-Sensor Fusion Technology, and Data Mining Algorithms in Cloud and Edge Computing-Based Smart City Environments." *Geopolitics, History, and International Relations*, vol. 14, no. 1, 2022, pp. 91–106.
- [19]. Zhu, H. "Research on the Application of Multi-Scale & Multi-Sensor Fusion Algorithm in MEMS Gyroscope Data Processing." *Journal of Physics: Conference Series*, vol. 2023, no. 1, IOP Publishing, Sept. 2021, p. 012039.
- [20]. Dong, J., et al. "Integration of Edge Computing and Blockchain for Provision of Data Fusion and Secure Big Data Analysis for the Internet of Things." *Wireless Communications and Mobile Computing*, 2022, doi:10.1155/2022/4627830.