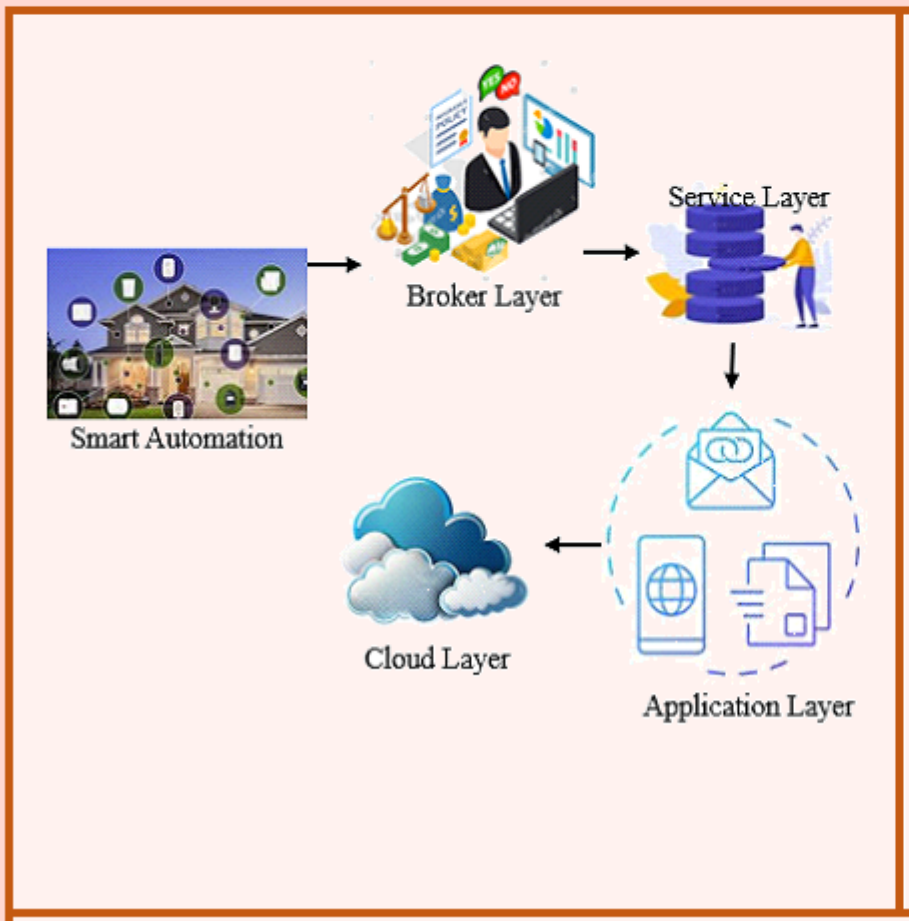
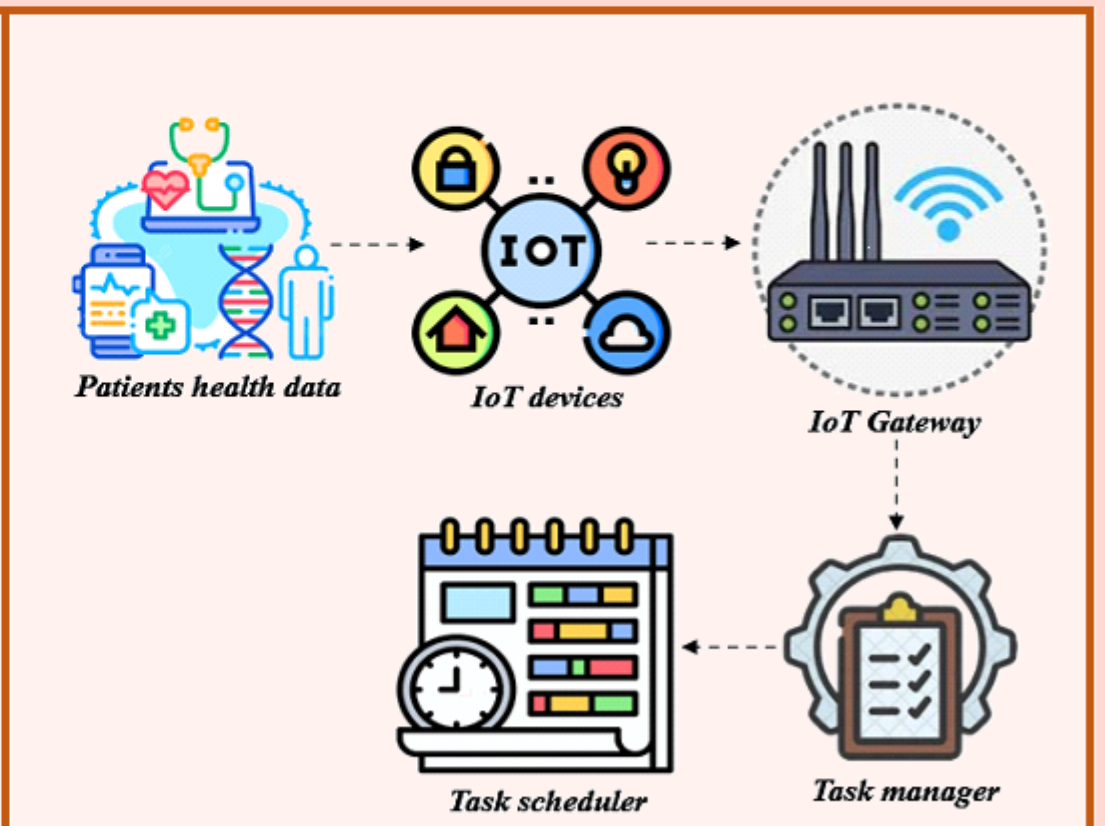
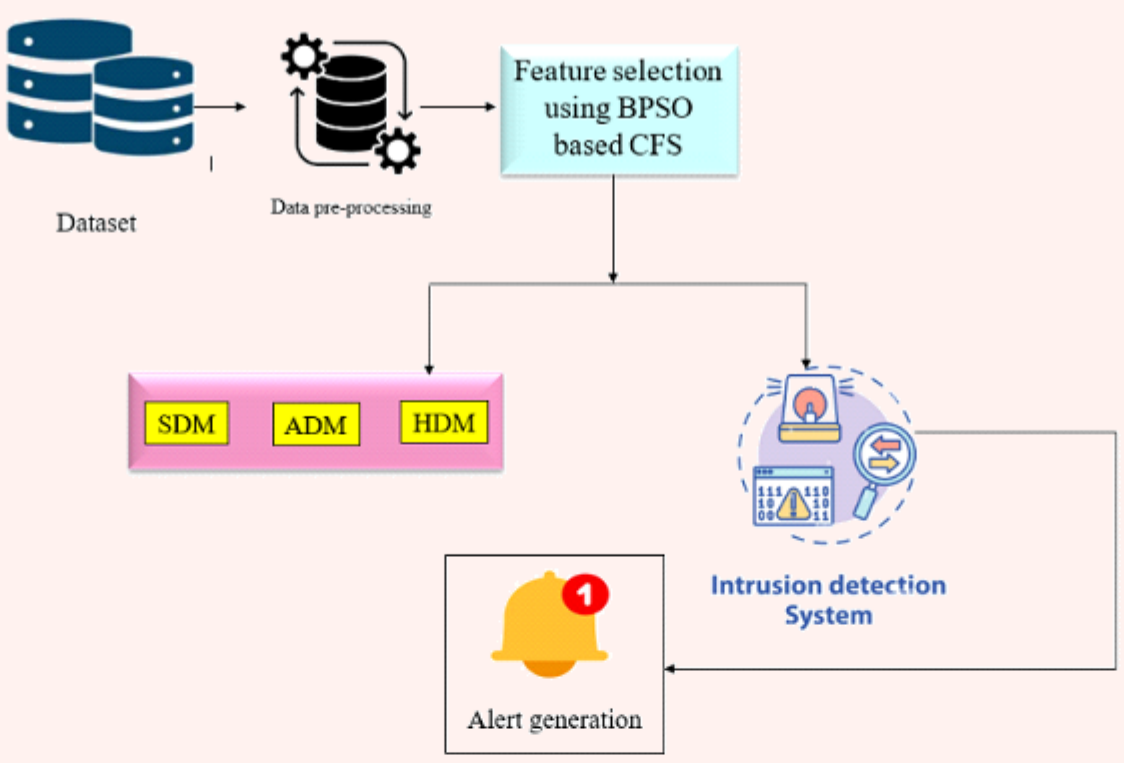
Volume 02, Issue 02, July - December (2024)

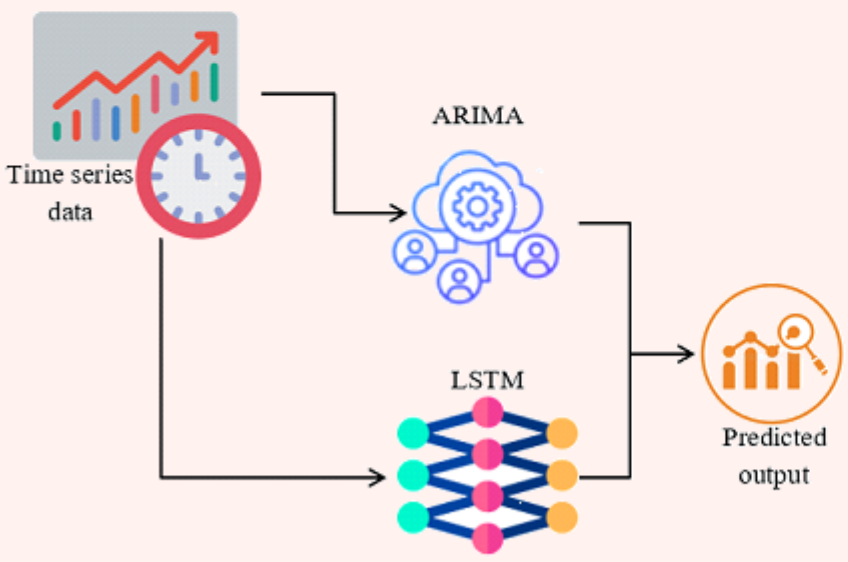
ISSN: 3048-801X

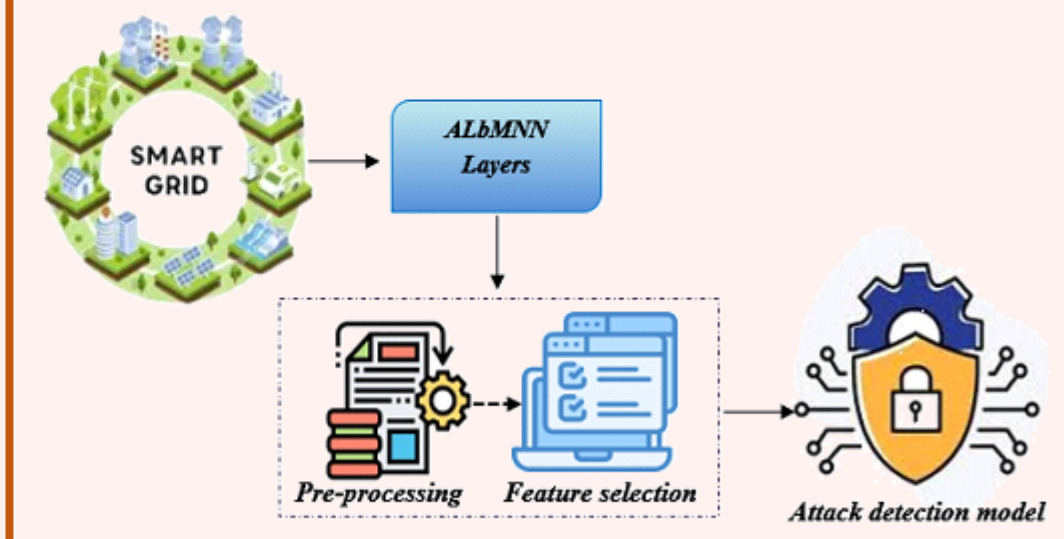
International Journal of System Design and Computing
IJSDC









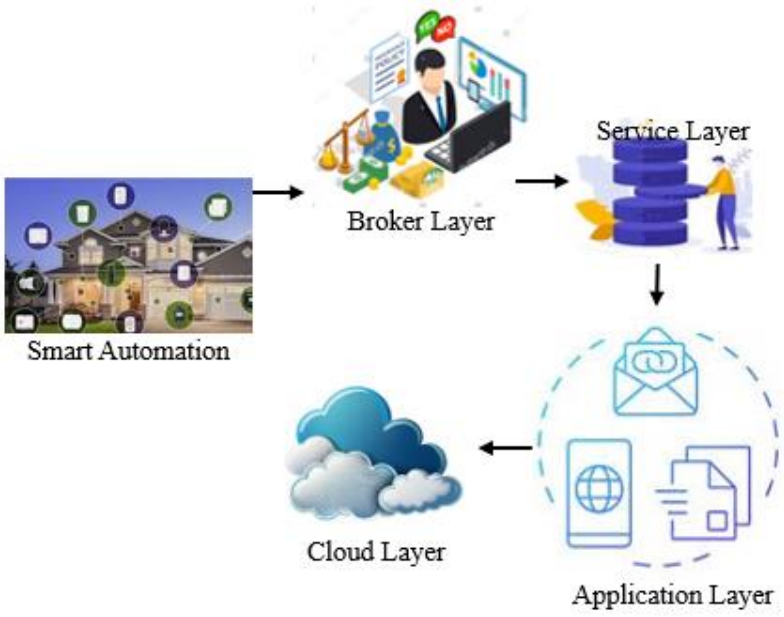


International Journal of System Design and Computing
IJSDC

1. COST-EFFICIENT RESOURCE MANAGEMENT IN SMART HOME AUTOMATION

J. Bharathi and C. Anuradha

Abstract – In this modern generation, there is a lot of technological improvement in Smart Home Automation. Real-time data collection and transmission through sensing devices is possible in the rapidly expanding "Internet of Everything" network of interconnected objects. The most realistic solution seems to be the affiliation of semantic information with all these diverse data sets via resource management, empowering their reusing and the implementation of rationale processes meeting up as well as skills for several momentary home automation control apps for a development's building providers of sub-standard systems lacking in intelligence. To overcome these issues in this paper, CREM has been proposed. In this research, a novel Cost-efficient Resource Management has been proposed in Smart Home Automation. The presented CERM has been simulated using MATLAB. A comparison is made between the suggested framework and current methods like VRP, SVM, PSO, and TBSA. Considerations include energy use, network longevity, bandwidth, latency, and response time. According to experimental findings, the proposed CERM technique by 39.75%, 25.725%, and 19.25% compared with VRP, SVM, PSO, and TBSA methods.

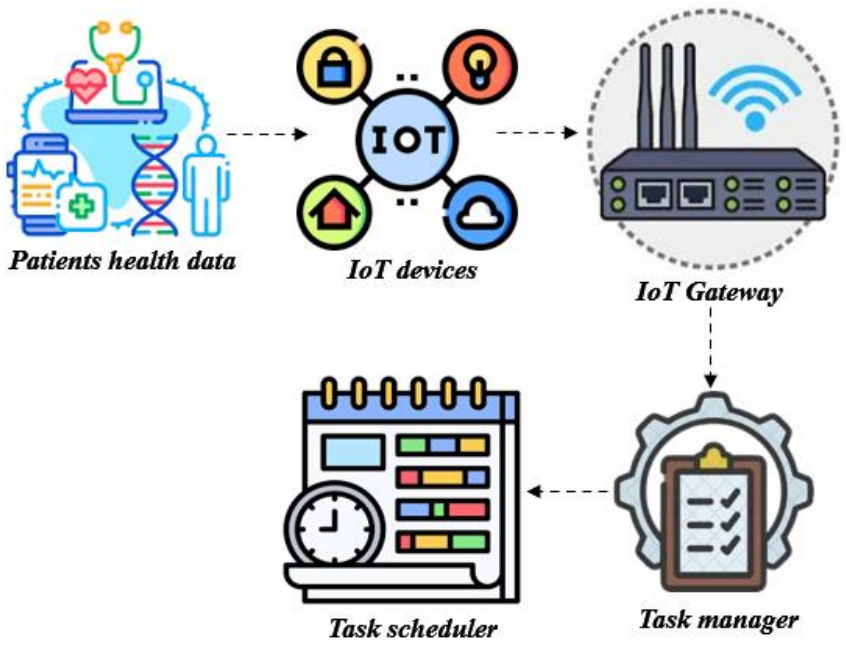


Keywords – Internet of Things, Constrained Application Protocol, Hidden Markov Model, Cost-effective Resource Management, Smart Home Automation.

2. DYNAMIC POWER ALLOCATION IN IOT-CLOUD ENVIRONMENT FOR HEALTHCARE APPLICATIONS

Kumarraja Andanapalli and M. Suresh Kumar

Abstract – In the healthcare industry, the integration of the Internet of Things (IoT) and cloud computing (CC) enables access and sharing of worldwide health datasets. Thus, it solves complex problems like data security, privacy, and storage. However, the wide usage of cloud infrastructure increases traffic and reduces cloud performance. Hence, in this article, a novel hybrid Honey Pot-based Feed Feedback Neural System (HPbFFNS) framework was proposed to allocate resources optimally to medical applications (tasks). This framework incorporates the features of Honey pot optimization, and Feed Feedback Neural Network (FFNN). Initially, the health information of patients is collected using the IoT devices and forwarded into the gateway layer for further processing. The task scheduler in the gateway layer analyzes the resource availability, deadline, and priority of the incoming requests to reduce the response time, and waiting time. The honey pot fitness function in the resource allocator helps to allocate resources optimally to the tasks. Additionally, for verification purposes, the results are contrasted with those of current techniques. The experimental and comparative analysis confirms that the suggested model outperforms the traditional algorithms in terms of response time, energy consumption, and resource utilization.

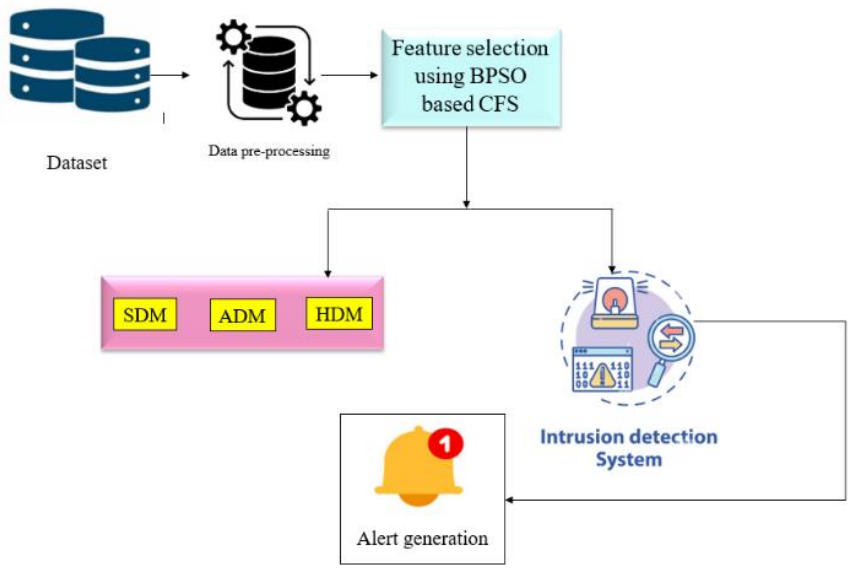


Keywords – Honey Pot Optimization, Feed Forward Neural System, Dynamic power allocation, Optimal Resource Allocation.

3. NETWORK INTRUSION DETECTION SYSTEM WITH AN EDGE BASED HYBRID FEATURE SELECTION APPROACH

A. Biju and S. Wilfred Franklin

Abstract – An intrusion detection system looks through network data to find both legitimate and malicious activity. This study can detect new attacks, which is especially useful in IoT situations. Deep Learning (DL) has demonstrated its superiority in solving challenging real-world issues such as NIDS. This method, however, necessitates more processing resources and takes a lengthy time. During a classification process, feature selection is critical in selecting the best attributes that best describe the goal concept. A novel Network intrusion detection (NEST) technique has been proposed to develop an improved edge-based hybrid feature selection approach, which is a deep learning method for detecting malicious traffic. The Enhanced BPSO technique overcomes the difficulty of BPSO feature selection by combining Binary Particle Swarm Optimization (BPSO) and correlation– based (CFS) traditional statistical feature selection. Three intrusion detection module having three classifiers make up the proposed system. The Signature Detection Module (SDM) examines threats and classifies it as unknown, normal, or intruder based on matching signatures utilizing the Generalized Suffix Tree (GST) algorithm. The Anomaly Detection Module (ADM) employs deep Q-learning to detect unknown attacks. The Hybrid Detection Module (HDM) employs the Meta- AdaboostM1 algorithm. Results of simulation show that the proposed protocol increases Detection Rate, computation time, and False Alarm Rate when compared to the existing XGBOOST-DNN, HNGEA, and HDLNIDS methods. The detection rate of the proposed NEST method is 4.55%, 6.47%, and 10.32% higher than the existing XGBOOST-DNN, HNGEA, and HDLNIDS, techniques respectively.

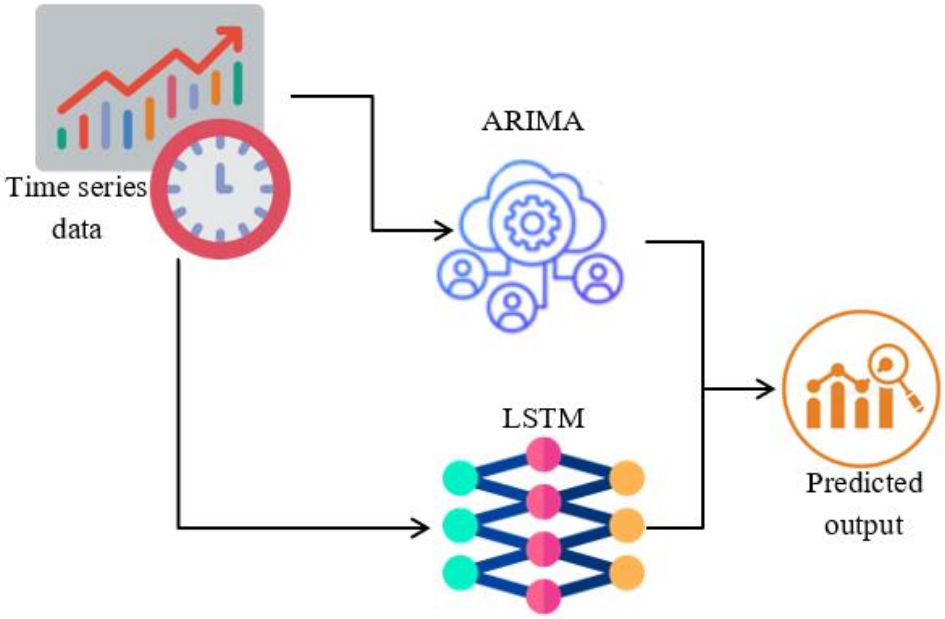


Keywords – Network Intrusion Detection System, Deep Learning, Generalized Suffix Tree, Q-learning, Meta-AdaboostM1.

4. PREDICTIVE MONITORING FRAMEWORK FOR ENHANCING OPERATIONAL RESILIENCE IN RETAIL FULFILLMENT SYSTEMS: A CASE STUDY

Ramya Thatikonda

Abstract – This paper presents the development and transformative impact of an innovative unified predictive monitoring framework designed for a large-scale retail fulfillment system. By seamlessly integrating cutting-edge industry-standard tools such as Grafana, Splunk, Kibana, and Elasticsearch, the framework provides unprecedented real-time visibility into fulfillment operations. The system leverages advanced machine learning algorithms, including time series forecasting, anomaly detection, and classification models, to proactively identify and resolve potential issues, particularly during high-demand periods. This data-driven approach has dramatically improved system stability, reducing service interruptions by 30% and enhancing customer satisfaction scores by 15%. The framework sets new benchmarks for operational resilience in the retail sector, demonstrating the transformative power of predictive analytics in managing complex fulfillment systems at scale.

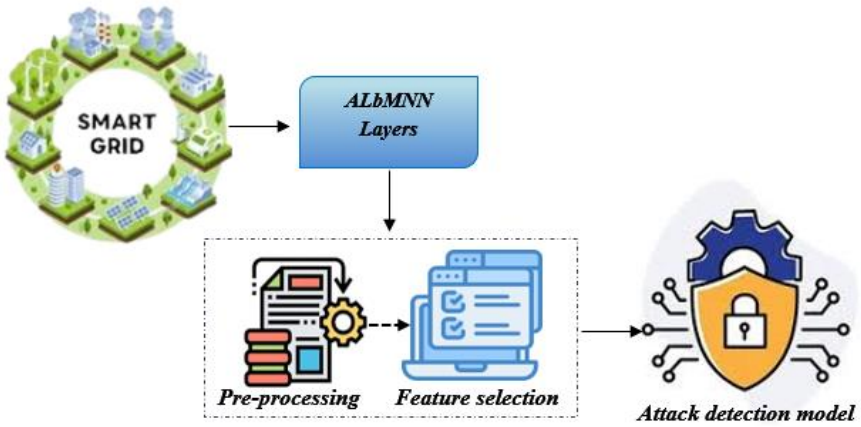


Keywords – Predictive Monitoring; Operational Resilience; Retail Fulfillment Systems; Machine Learning.

5. HYBRID NEURAL SYSTEM FOR CYBER-ATTACK DETECTION IN LARGE-SCALE SMART GRIDS

R.A. Mabel Rose and V. Padmajothi

Abstract – The advancements in the distributed energy system and digital technology in the smart grid system increase efficiency, stability, and reliability. However, it increases the vulnerabilities in the grid network. The falsely injected data in the grid network leads to failures in energy production, and consumption. Hence, an Ant Lion-based Modular Neural network (ALbMNN) model was proposed to detect the normal and malicious data. The presented model integrates the ant lion fitness and MNN attribute to detect the falsely injected data in the grid system. The dataset was initialized and pre-processed using the divide-and-conquer principle of MNN. The optimal ant lion fitness solution helps in selecting features optimally from the dataset. Finally, the presented model was assessed with a large-scale smart grid dataset, and the results are estimated. Moreover, a comparative analysis was performed to verify the performance of the developed scheme. Based on performance and comparative analysis, the suggeated model performed better than other existing methods.



Keywords – False Data Injection, Cyber-attack detection, Smart grids, Modular Neural Network, Ant Lion Optimization.

International Journal of System Design and Computing (IJSDC)
Volume 02, Issue 02, July - December (2024)

RESEARCH ARTICLE

COST-EFFICIENT RESOURCE MANAGEMENT IN SMART HOME AUTOMATION

J. Bharathi [1, *] and C. Anuradha [2]

[1] Department of Electronics and Communication Engineering, Deccan College of Engineering and Technology, Hyderabad, Telangana 500001 India.
[2] Department of Electrical and Electronics Engineering, SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu 603203 India.
[*]*Corresponding e-mail: bharathi@deccancollege.ac.in*

Abstract – In this modern generation, there is a lot of technological improvement in Smart Home Automation. Real-time data collection and transmission through sensing devices is possible in the rapidly expanding "Internet of Everything" network of interconnected objects. The most realistic solution seems to be the affiliation of semantic information with all these diverse data sets via resource management, empowering their reusing and the implementation of rationale processes meeting up as well as skills for several momentary home automation control apps for a development's building providers of sub-standard systems lacking in intelligence. To overcome these issues in this paper, CREM has been proposed. In this research, a novel Cost-efficient Resource Management has been proposed in Smart Home Automation. The presented CERM has been simulated using MATLAB. A comparison is made between the suggested framework and current methods like VRP, SVM, PSO, and TBSA. Considerations include energy use, network longevity, bandwidth, latency, and response time. According to experimental findings, the proposed CERM technique by 39.75%, 25.725%, and 19.25% compared with VRP, SVM, PSO, and TBSA methods.

Keywords *– Internet of Things, Constrained Application Protocol, Hidden Markov Model, Cost-effective Resource Management, Smart Home Automation.*

1. INTRODUCTION

Internet of Things is a network of embedded devices that can connect to the internet and exchange data with one another. Different sensors and actuators are employed by IoT networks to create intelligent systems [1]. As new emerging technologies rise, the contribution of IoT to care has become increasingly important. As a result, automation will be required across every field in the near future. Transportation gridlock, disposal of waste, means the greater, logistics, security, control systems, emergency services, and healthcare are all improved by IoT technologies [2].

As technology has developed in recent years, human-machine interaction (HMI) has become increasingly practical in daily life. The Internet, which was previously utilized for interaction but was also used for objects, i.e., the Internet of Things, has been used to further HMI research in recent years. Connecting anything to the Internet so that it may be accessed from anywhere is the aim of this application [3].

A smart home automation system gives customers control over a number of frequently used items., as well as one that makes doing so much simpler and more energy-efficient. These days, automation systems for buildings and homes are employed often [4]. More people today are conscious that they must make their houses environmentally friendly. Smart houses assist users save money and energy by managing irrigation, lighting, window coverings, and usage [5].

Resource management, rather than the way of coordinating and allocating resources to application fields, was the main principle of shared databases [6]. A high-quality service while reducing energy consumption to reduce household electricity costs is achieved by managing variables with sophisticated multi-objective decision-making. On networks with more than 250 systems, slower transmit cycles (CoAP) are essential for resource discovery. By using MQTT, you can subscribe to topics in a flexible manner. Security issues with encryption on a large scale [7].

In this paper, a novel proposed CERM-SHA uses a COAP protocol for security into account when dealing with IoT protocols The most effective solution seems to be the affiliation of semantic information with all these diverse data sets via resource management, empowering their reusing and the implementation of rationale processes meeting up as well as skills for several momentary home automation control apps for a development's building providers of sub-standard systems lacking in intelligence [17-19]. To overcome these issues in this paper, CREM has been proposed.

The remaining portion of the paper is arranged as follows: The Literature Survey is described in Section 2. Section 3 describes the proposed system. Section 4 presents the results and discussion. Conclusion and future work are described in Section 5.

2. LITERATURE REVIEW

The Internet of Things (IoT), which refers to actual objects with sensors, processing power, software, and other techniques that exchange data and interact with other systems and devices over the Internet or another communications network, encompasses the topic of resource management in this study.

In 2018, Al-Kuwari, et al. [8] have been proposed a solar-house monitoring system. The proposed system is managed by the same IoT platform that communicates to EmonCMS immediately via NodeMCU. There is a problem with the NodeMCU board having just one analog input, which restricts its use to that of a single system for monitoring data.

In 2021, Yar, et al. [9] have suggested a low-cost, complete smart home system built on the principles of edge computing and the IoT. The proposed solution consists of edge computing stores sensitive data on a customer's local cloud to preserve their privacy One disadvantage The disadvantage of wireless home automation systems is also that they can only be utilized in a single room indoors or within the scope of the Bluetooth signal. In comparison to cutting-edge methods, the proposed system detects movement 5% faster and switches the relay around within 5 and 4 minutes, respectively. Additionally, it is 6% more energy-efficient than the alternatives now in use.

In 2019, Gill, et al. [10] suggested a resource management method that uses PSO to optimize many resources at once for fog-enabled cloud computing systems. According to the survey, implementing us technique will reduce the bandwidth of the network by 10%, reaction time by 12%, latency by 12.35%, and energy consumption by 14%.

In 2016, Pirbhulal, et al. [11] has been suggested a Triangle Based Security Algorithm (TBSA), which is based on an efficient key generation mechanism. TBSA was integrated into WSNs and the Internet to create a ground-breaking. IoT-based smart house capable of secure data transmission more than long-distance across all various relevant sensor nodes within the network. Based on the results of hardware design, the proposed algorithm TBSA is far more effective in every other aspect of cryptography.

In 2018, Jabbar, et al. [12] has been suggested a low-cost, Wi-Fi-based Smart Home (SH) automation system with an Android app that enables remote device management and monitoring. This study proposes, but does not plan, a simplified Wi-Fi-based Automation System for Smart Home prototype using an Arduino and an Android smartphone.

In 2020, Majeed, et al. [13] proposed a novel concept of home automation that employs a machine-learning method for smart decision making, as well as blockchain, which are used to ensure IoT device verification and identity. Sensor data such as temperature, smoke, and lamp are included in the input dataset. The status classes of the challenging task, such as "ON" and "OFF." creates status classifications, such as "ON" or "OFF." A value of y = 0 means the device is "OFF," while a value of y = 1 means the device is "ON." When y = 0, the gadget is said to be "OFF," and when y = 1, it is said to be "ON."

In 2019, Jabbar, et al. [14] Proposed an IoT@HoMe offering a low-cost perfect blend (local and remote) Sensor home automation system with a consumer interaction for laptops and smartphones. An IoT-based home-automation system may conveniently and inexpensively operate devices. So, via the Internet, to improve home security through automatic control. The suggested IoT@HoMe system also contrasts sharply with the current arrangement to highlight the key features.

In 2017, Yadav, et al. [15] suggested a versatile, affordable, and multipurpose smart home monitoring and management solution. The plan is working. Thanks to the node-MCU ESP32, this gadget may be controlled remotely and has access to the Internet. It transmits sensor data to the Firebase database and is capable of receiving commands from the server, which enables automated control. The proposed system will provide greater flexibility for dealing with things automatically. This will reassure the user of improved security. The current system has many limitations, such as the issue of energy consumption, water waste, child safety, and so on.

In 2019, Sarmah, et al. [16] proposed a safe and efficient smart home system capable of protecting homes from theft or unusual activity while saving power. The system is evaluated using upon which at KU college, analysing 30 apartment buildings for 60 days and exploring people to be amazingly beneficial in terms of thieves' safety and electricity savings when compared to existing systems.

In 2018, Rout, et al. [17] proposed SHAS prototype along with its Android App has been successfully implemented. proposed technology demonstrator smart home scheme, including an Android app and an ATmega16 central controller, has been effectively applied, and the results are presented. it can control the status of the fan or AC, according to the humidity and temperature of the room.

3. PROPOSED SYSTEM

In general, IoT devices gather and send information to Resource Management (RM), which can process it. As Connected systems and servers proceed to interact, large amounts of data are transmitted through IoT networks, causing significant operational costs and potential concerns in the network. The proposed work investigates the hidden Markov (HMM) prototype sequential data generated by IoT devices. HMM, functions are processed at Resource Management. The findings are generated by HMM's hidden states. HMM is an excellent tool for trying to capture and model string sequences. The great powers in HMM are enclosed and non - observable.

Device Layer

The devices layer, also known as the input nodes, ought to be the initial area of the proposed models. In this, different kinds of variable and graphical sensing devices are used for automation, safety, and safety. They also help save energy inside the smart home by enclosing the environment for door verification and particle sensors like fingerprint images. This

implementation incorporates low-level sensors into the smart home system to gather environmental information. This same lamp is controlled by a similar Vibration motor that detects movement within the house. The thermostat sensor uses the temperature of the bed to control the cooling and heating systems. If the average temp exceeds a present threshold and there is motion, the air conditioning system will automatically turn on; otherwise, the air conditioning unit will turn off.

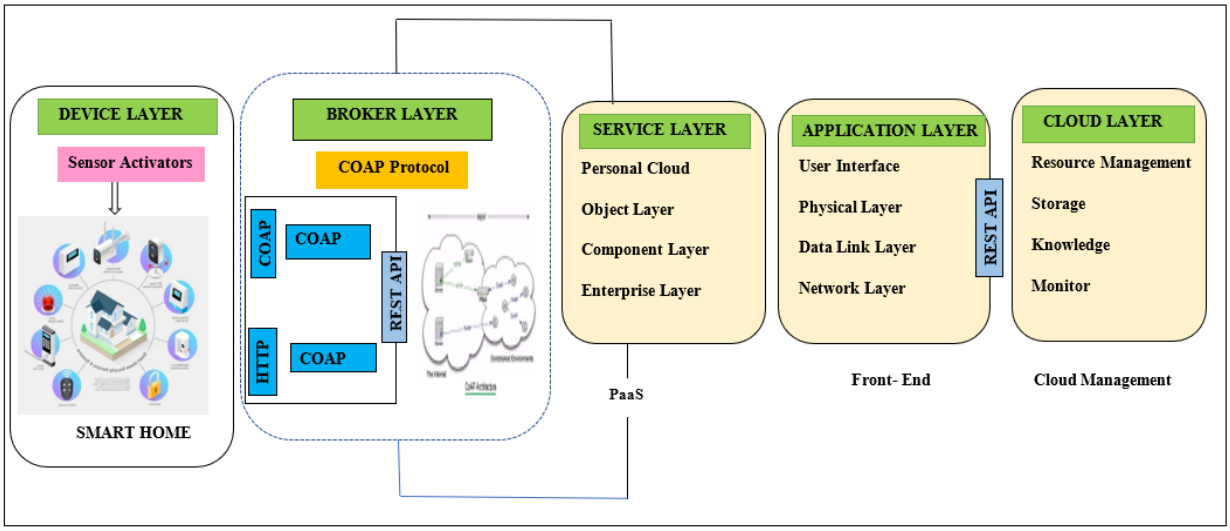


Figure 1. Proposed Hidden Markov Model for efficient CoAP Protocol

The intensity of sunlight is measured using photoresistor sensors. The house's outdoor lighting program basically turns on if the number of lights goes below a predefined level, and vice versa. Devices which measure water are used to estimate the volume of water in a tank. When the water level drops below a certain point, the liquid engine immediately starts., while when the liquid tank is full, the liquid engine turns off automatically.

The possibility to take up smoking also was identified, in addition to Liquefied petroleum gas flow in the restaurant and residences. As an alternative, flame sensors are utilized to locate flames in the region. The siren activates and an SMS alert is issued to the designated officer when gas, smoke, or fire are discovered. Security has improved, and the data set is safeguarded from unauthorized access using a proper authentication strategy to preserve system confidentially. The graphical sensor is used to stream video to homeowners' laptops and cell phones.

Broker Layer

The Constrained Application Protocol (CoAP) protocol is used by the broker layer to transmit commands and data from many sensors to the server side. The operating principle of CoAP protocols is depicted in Figure 1. The REST and CoAP protocols comprise the broker layer. Representational State Transfer (REST) is commissioned by the European on which Rest API services are based. REST is a software system that makes filled use of all HTTP characteristics such as browser connectivity, expandability, verification, and encoding.

Service Layer

The layer is in charge of monitoring the brokerage layer's data receipt when RPI is only utilized as a portal service. The Node-JS dashboard is configured at the service layer. Other layers are served by this cross-layer utility. Subscribers throughout Node-JS receive messages for topics that they have signed either through Node-JS, this completely carries out these publication strategies. The nodes that can send messages are found in Node-JS. JSON and HTML files were stored within a Node-JS-Flow node. The node connects to other nodes for sharing of information and provides a frictional pressure visual web-based editor. The flow designer can be accessed via an internet browser at http://ipaddress:1880. (Retrieved July 10, 2021) Node-JS offers inputs, outputs, functionality, and community nodes. Within such processes, we can switch on and off our equipment, monitor the state of several sensors, and keep an eye on the surroundings. In the Node-JS process, enabling a device is as basic as simply dragging and dropping ends, such as by allocating a GPIO (General Purpose Input and Output) pin. Node-JS also allows for rapid scaling. The service layer also controls software, data, private clouds, and data aggregation. We were storing data via two different methods of stockpiling: a device called My-Personal cloud device and cloud storage. A simple-to-use personal storage device called My-Personal cloud plugs straight into a Wi-Fi router at home to store all visual data in one area while the cloud provides cloud storage. Cloud computing refers to the exchange of cloud computing such as apps, servers, networking, and so on. Cloud computing, for example, makes use of the Internet to provide storage, information access, and computer resources. Cloud deployment models that are most commonly used are IaaS, SaaS, and PaaS. Data aggregation, software administration, personal cloud, and data management are the four main characteristics of the service layers.

Application Layer

Using a user-friendly design, Specific IoT management front-end applications are developed using the developer

portal. The user interface for viewing and managing multiple Node-JS gadgets is implemented in this layer. The end-user application is designed to dynamically display sensor data and instantaneously control electronic equipment when the change takes place. The front-end implementation is examined to confirm appropriate operation of the remote and automated electronic component control systems. The Node-JS base station analyses alterations in the environment and analyses how electrical equipment reacts to those alterations. The controller must function in a way that satisfies the necessary requirements and react swiftly to environmental changes. In addition, the remote control must react immediately to the right device. Applications thus need to be responsively designed. Today, the large bulk of commonly produced web-based apps adhere to this criterion. Users of this system frequently want to use the internet software to manage their household control system from a smartphone. Type equation here. To connect a phone to the control centre over the internet, users must verify themselves in the application level.

Cloud Layer

A large amount of unstructured data is produced in a home setting by the multiplication of sensors and actuators at the device layer, this could be used to extract connections and important information. Frequently, there isn't enough storage to accommodate all of the input data. To extract significant and helpful information for coordinating local and household services, centralized storage is necessary. Using cloud services and a Simple Conscious Experience cloud device, we save personal information in a suitable database management system. In order to store the sensor data for future research, we average it at the conclusion of the era and transfer it to a cloud server.

Hidden Markov Model in Resource Management

HMM is a statistics theory that describes how inner, superficially visible processes result in compared to goals. We named the information a "symbol" and the unobserved heterogeneity component producing the measurement a "state"; together, these two unpredictable events, one unseen with hidden units while the other apparent with visible symbols, combine to form an HMM. The underpinning state causes the posterior distribution of the visible sign, when the Markov chain is made up of state variables. In light of this, alternative term for an HMM is a doubly-embedded stochastic process**.**

$$M\{b_{l+1} = i|b_l = j, b_{l-1} = j_{l-1}..., b_1 = j_1\} = M\{b_{l+1} = i|b_l = j\} = s(j, i) \quad (1)$$

For every state j, each is, and every l 1. The likelihood of transitioning from node j to state I represented by the symbol s, is known as the transition probability (j,i). We write the possibility of that nation for the original state b1 as (j)= M b1 = j for all j S. The underlying state bl alone determines the likelihood that the nth observation will be al = an, so

$$M\{a_l = a|b_l = j, b_{l-1}, a_{l-1}, ...\} = M\{a_l = a|b_n = j\} = d(a|j) \quad (2)$$

For every chance to try something new, every j opportunity, and every l 1. This is referred to as the emissions probability of x at the state I is represented by the representation d (a | j). An HMM has completely specified by the three probability measurements s(j), I(j), and d(a|j). For simplicity, we'll call this collection of parameters Q.

$$M\{a, b|Q\} = M\{a|b, Q\}M\{b|Q\} \quad (3)$$

Where

$$M\{a|b, Q\} = d(a_1|b_1)\, d(a_2|b_2)\, d(a_3|b_3) \ldots d(a_n|b_n) \quad (4)$$

$$M\{b|Q\} = \pi(b_1)\, s(b_1, b_2)\, s(b_2, b_3) \ldots s(b_{n-1}, b_n) \quad (5)$$

As we can see, trying to calculate the monitoring possibility is simple when we know the quality of the soil sequence.

Basic Problems and Algorithms for HMMS

Three principal concerns must be resolved before HMMs can be used in real world applications. Consider a new symbol sequence a=a 1a 2a 3....a n. How can we measure the monitoring chances based on a given HMM? Mleftx |Are you sure? This problem is also known as the goals scored problem because calculate the possibility Mlefta|0right is a natural way of'scoring' a testable theory sequence based on the model in question. It's important to remember that the underlying state sequence for a given an isn't readily observable, and multiple state sequences may exist resulting. As a result, one method for calculating the monitoring Probability is calculated by adding all possible case sequences b for the given a.

$$M\{a|10\} = \sum_b M\{a, y|10\} \quad (6)$$

But this requires a lot of processing power because there are numerous future governments. As a result, we urgently need a more effective way to calculate Mleftx |Qright. The forward algorithm in dynamic programming is capable of quickly calculating Mleftx |Qright. The following forward variable is defined by this algorithm rather than all possible state combinations.

$$\alpha(l, j) = M\{a_1 \ldots . a_l, b_l = j|Q\} \quad (7)$$

The computation of this parameter could be performed out repeatedly by employing the same theoretical calculation.

$$\alpha(a, j) = \sum_z [\alpha(l-1, c)t(c, j)d(a_l|j)] \quad (8)$$

For $l = 2 \ldots n$. We can calculate Ma|Q = c (N, c) at the end of the recursions. With only Q(NP2) computations, this technique computes the observation probability of a as a result, instead of increasing exponentially increasing sequences unit Length, the computation of probability grows just quadratic.

Identifying the optimal path, or ideal state sequence, for the HMM that increases the detection probability of the provided symbol sequence x is another real-world problem. People are interested in discovering the governmental order that adequately describes the measurable symbol series among all potential state action sequences y. It is commonly referred to as the optimal alignment issue since this can be seen as determining the best alignment here between phase is the process as well as the HMM. Formally, we are looking

for the optimum path y* that satisfies the conditions listed below.

$$b^* = xuf \max_b M\{a|a,Q\} \quad (9)$$

As we have Ma, b | @, this is the same as finding the maximizes state sequence Ma, b |. $M\{b|a,Q\} = \frac{M\{a,b|Q\}}{M\{a|Q\}}$ (10)

It's indeed practically difficult to compare every Pn alternative action patterns to get the "ideal" series b*. However, we may effectively figure out the best path b* to use a different dynamic computing algorithm called the Viterbi algorithm [14, 15]. The parameter is determined via the Viterbi algorithm.

$$\gamma(l,j) = \max_{b_1,\ldots,b_{n-1}} M\{a_1 \ldots a_l, b_1 \ldots b_{l-1} b_l = j|Q\} \quad (11)$$

Finally, we can calculate the maximum observation as follows

$$M^* = \max_b M\{a,b|Q\} = \max_c \gamma(n,c) \quad (12)$$

By going back through the nested loops that resulted in the greatest likelihood M = Ma, b|Q, identifying the best path y* is simple. The Viterbi algorithm, like the backward approach, discovers the best state sequence in Q(NP2) time.

The Viterbi algorithm, as demonstrated repeatedly, determines the best course to take in order to maximise the probability that its full symbol sequence would be observed. Finding the best states for each symbol position may be more beneficial in particular circumstances. In this situation, we may determine the best state, bl, that is the most like to be the condition that underlies al as follows.

$$\widehat{b_l} = x\ \gamma f \max_j M\{b_n = j|a,Q\}, \quad (13)$$

according to the supplied a and Q The post-hoc likelihood Mbl = j | a, Q can be calculated from

$$M\{b_l = j|a,Q\} = \frac{M\{a_1 \ldots . a_l, b_l = j|Q\} M\{a_{n+1} \ldots . a_n | b_l = j, Q\}}{M\{a|Q\}} \quad (14)$$

$$= \frac{\alpha(l,j)\beta(l,j)}{\sum_c \alpha(l,c)\beta(l,c)} \quad (15)$$

The backwards algorithm, as shown below, allows for a cyclical determination of this backward variable (l,j).

$$\beta(l,j) = \sum_c [t(j,c) d(a_l|c) \beta(l+1,c)] \quad (16)$$

where n = L-1, L-2,..., 1. Making distinct forecasts for ideal states maximises the expected number of correctly predicted states, which is a benefit. Px, y |Px, y| will result from the overall state sequence, y=y1y2...yL, which is less than optimal. When this occurs, Px, y|=0 since there is a chance that the projected path y isn't even a valid path in the given HMM. Due to this, the posterior-decoding method is typically chosen when our main concern is forecasting the best outcome at a fixed position rather than determining the best state sequence for the entire monitoring interval. You may estimate a government's dependability using the posterior probability in. For instance, after using the Viterbi method to find the best course of action y=y1...yL, we may compute the prior distribution Pyn=yn|x to determine the dependability of each state prediction yn. Using the provided HMM, both scoring and alignment issues analyse a new observation sequence. But only if the HMM can faithfully represent the sequence in concern would the answers to such issues be meaningful. Consider if people obtain a set of related inspection sequence alignments A = a1, a2, as, which we want to describe using an HMM. They could be distinct speech samples containing the same word or protein structures from the same stable family. The real question now is how to select the HMM variables in a rational way based on these observational data. This is usually referred to as the learning problem. Even though no appropriate way exists for assessing variables from a limited amount of tracking action scenes there are methods to locate the HMM variables that maximise the regional observation probability. The Baum-Welch learning algorithm and the HMM are two examples. Algorithm Baum-Welch is a forward-backward assumption (EM) method that assumptions and updates values iteratively. People can also employ basic gradient-based methods to find the best HMM. variable in this kind of situations even though assessing the HMM variables is largely an optimization., The Monte Carlo EM (MCEM) technique, that also uses the Monte Carlo strategy for estimating the Prediction algorithm's so-called E-step (expectation step) can be used to train the HMM. There are also probability optimization-based training techniques, such as evolutionary algorithm, that attempt to improve results by avoiding local maxima. For any further discussion of the estimation of parameters for concealed Markov models, this same viewer is guided to this published research.

4. RESULT AND DISCUSSION

In the result section evaluates the proposed technique with the existing techniques VRP, SVM, PSO, and TBSA, methods. The simulation of every existing approach and the proposed model is carried out using MATLAB R2020b.

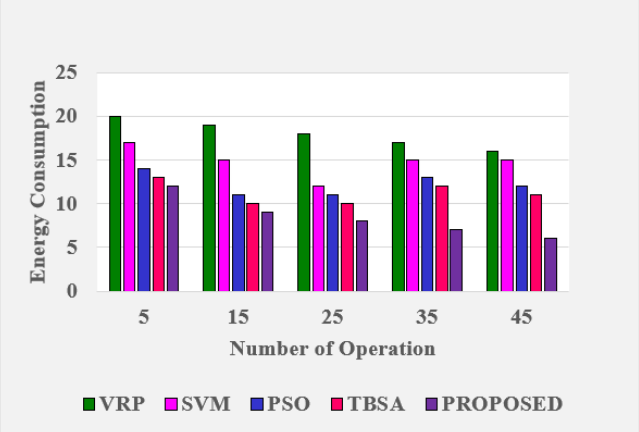


Figure 2. Number of Operation Via Energy Consumption

Figure 2 displays the energy usage for various node counts and data transmission rates. These variables are examined and compared to the existing VRP, SVM, PSO, and TBSA. Through active-sleep state switching, the proposed technique maintains a significant number of energy-efficient nodes for use in future transmission. This prevents any node from ever being used and depleted. This lowers overall energy consumption because not all nodes need to use energy for a predefined transmission.

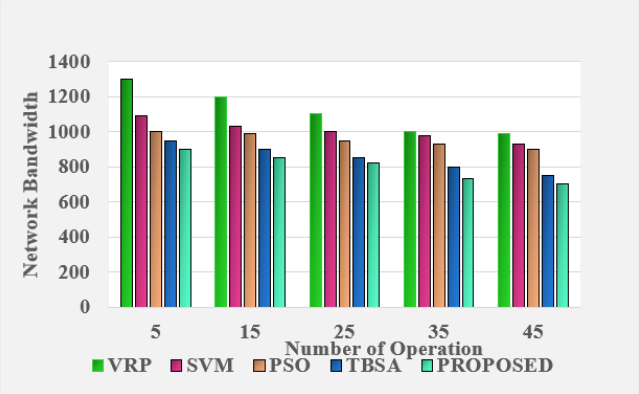


Figure 3. Number of Operation via Network Bandwidth

Figure 3 compares the network bandwidth consumption across different methods VRP, SVM, PSO, TBSA, and a Proposed method over varying numbers of operations. VRP consistently shows the highest bandwidth usage, particularly at lower operation counts, while the Proposed method demonstrates the lowest bandwidth consumption across all operations, indicating its efficiency. SVM, PSO, and TBSA exhibit moderate bandwidth usage, with a gradual decrease as the number of operations increases. This analysis highlights the Proposed method is superior performance in minimizing network bandwidth, making it a more scalable and efficient solution as operations grow.

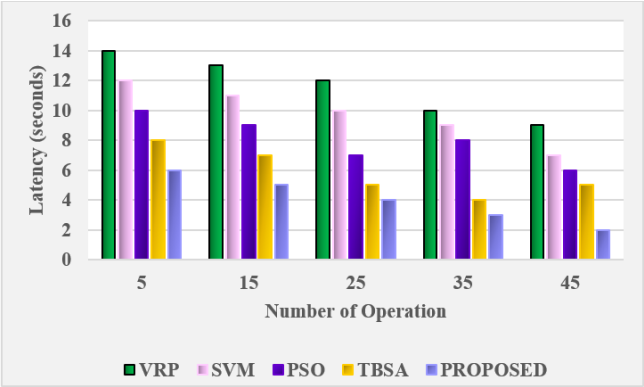


Figure 4. Number of Operation Via Latency

Figure 4 compares the Latency analysis to alternative approaches. The proposed, HQCA, VRP, SVM, PSO, and TBSA, respectively. Despite this, the proposed technique has been able to reduce the Latency, proving its feasibility with fluctuating node counts.

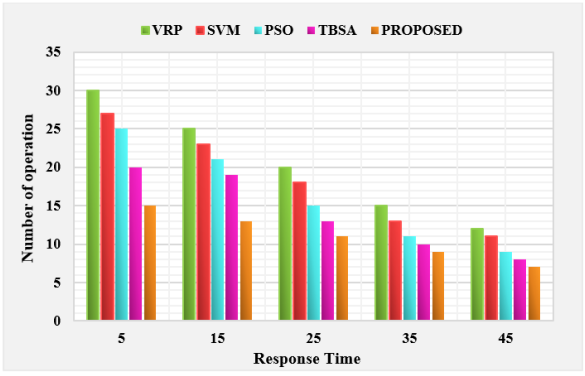


Figure 5. Response Time Via Number of Operations

Figure 5 depicts the performance of five approaches VRP, SVM, PSO, TBSA, and a proposed method based on the number of operations completed for varying response times (5 to 45 units). The proposed method consistently shows a higher number of operations across all response times, indicating superior efficiency and adaptability compared to the others, especially at lower response times, where its advantage is more pronounced.

5. CONCLUSION

In this paper CERM has been proposed a lot of technological advancement in Smart Home Automation inside this modern generation. The term "IoT" describes a rapidly growing network of linked devices that have embedded sensors that allow them to gather and send data in real time. The best solution appears to be the association of semantics with these heterogeneous data through resource management, allowing them to be reprocessed and reasoning mechanisms to be executed. We proposed a novel Proposed Cost-Effective Resource Management in Smart Home Automation throughout this paper (CERM). MATLAB was used to simulate the presented CERM. The conceptual methodology is compared to existing techniques including such VRP, SVM, PSO, and TBSA in regards in terms of network lifetime, network bandwidth, latency, and response time. According to experimental data, the proposed CERM method outperformed the VRP, SVM, PSO, and TBSA methods by 39.75%, 25.725%, and 19.25%, respectively. The progress will see a gain in the latest equipment and technology along with appliances, reducing manual every aspect of smart home automation.

CONFLICTS OF INTEREST

Not applicable.

FUNDING STATEMENT

Not applicable.

ACKNOWLEDGEMENTS

The author would like to express his heartfelt gratitude to the supervisor for his guidance and unwavering support during this research for his guidance and support.

REFERENCES

[1] D.A. Gandhi, and M. Ghosal, “Intelligent healthcare using IoT an extensive survey,” In *2018 second international conference on inventive communication and computational technologies (ICICCT)*, pp. 800-802, 2018. IEEE. [CrossRef] [Google Scholar] [Publisher Link]

[2] C. Stolojescu-Crisan, C. Crisan, and B.P. Butunoi, “An IoT-based smart home automation system,” *Sensors*, vol. 21, no. 11, pp.3784, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[3] S.K. Vishwakarma, P. Upadhyaya, B. Kumari, and A.K. Mishra, “Smart energy efficient home automation system using IoT,” In *2019 4th international conference on internet of things: Smart innovation and usages (IoT-SIU)*, pp. 1-4, 2019. IEEE. [CrossRef] [Google Scholar] [Publisher Link]

[4] U. Singh, and M.A. Ansari, “Smart home automation system using Internet of Things,” In *2019 2nd International Conference on Power Energy, Environment and Intelligent Control (PEEIC)*, pp. 144-149, 2019. IEEE. [CrossRef] [Google Scholar] [Publisher Link]

[5] T.S. Gunawan, I.R.H. Yaldi, M. Kartiwi, N. Ismail, N.F. Za'bah, H. Mansor, and A.N. Nordin, "Prototype design of smart home system using internet of things," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 7, no. 1, pp.107-115, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[6] M.J. Iqbal, M.M. Iqbal, I. Ahmad, M. Ahmad, N.Z. Jhanjhi, S. Aljahdali, and M. Mushtaq, "Smart Home Automation Using Intelligent Electricity Dispatch," *IEEE Access*, vol. 9, pp.118077-118086, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[7] J.T.S. Kim, "Analyses of Open Security Issues for Smart Home and Sensor Network Based on Internet of Things," *IoT Appl. Computer*, pp.179-196, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[8] M. Al- Kuwari, A. Ramadan, Y. Ismael, L. Al-Sughair, A. Gastli, and M. Benammar, "Smart-home automation using IoT-based sensing and monitoring platform," In *2018 IEEE 12th International Conference on Compatibility, Power Electronics and Power Engineering (CPE-POWERENG 2018)*, pp. 1-6, 2018. IEEE. [CrossRef] [Google Scholar] [Publisher Link]

[9] H. Yar, A.S. Imran, Z.A. Khan, M. Sajjad, and Z. Kastrati, "Towards smart home automation using IoT-enabled edge-computing paradigm," *Sensors*, vol. 21, no. 14, pp.4932, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[10] S.S. Gill, P. Garraghan, and R. Buyya, "ROUTER: Fog enabled cloud based intelligent resource management approach for smart home IoT devices," *Journal of Systems and Software*, vol. 154, pp.125-138, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[11] S. Pirbhulal, H. Zhang, M.E. Alahi, H. Ghayvat, S.C. Mukhopadhyay, Y.T. Zhang, and W. Wu, "A novel secure IoT-based smart home automation system using a wireless sensor network," *Sensors*, vol. 17, no. 1, pp.69, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[12] W.A. Jabbar, M.H. Alsibai, N.S.S. Amran, and S.K. Mahayadin, "Design and implementation of IoT-based automation system for smart home," In *2018 International Symposium on Networks, Computers and Communications (ISNCC)*, pp. 1-6, 2018. IEEE. [CrossRef] [Google Scholar] [Publisher Link]

[13] R. Majeed, N.A. Abdullah, I. Ashraf, Y.B. Zikria, M.F. Mushtaq, and M. Umer, "An intelligent, secure, and smart home automation system," *Scientific Programming*, 2020, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[14] W.A. Jabbar, T.K. Kian, R.M. Ramli, S.N. Zubir, N.S. Zamrizaman, M. Balfaqih, V. Shepelev, and S. Alharbi, "Design and fabrication of smart home with internet of things enabled automation system," *IEEE access*, vol. 7, pp.144059-144074, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[15] V. Yadav, S. Borate, S. Devar, R. Gaikwad, and A.B. Gavali, "Smart home automation using virtue of IoT," In *2017 2nd International Conference for Convergence in Technology (I2CT)*, pp. 313-317, 2017. IEEE. [CrossRef] [Google Scholar] [Publisher Link]

[16] R. Sarmah, M. Bhuyan, and M.H. Bhuyan, "SURE-H: A Secure IoT Enabled Smart Home System," In *2019 IEEE 5th World Forum on Internet of Things (WF-IoT)*, pp. 59-63, 2019. IEEE. [CrossRef] [Google Scholar] [Publisher Link]

[17] K.K. Rout, S. Mallick, and S. Mishra, "Design and implementation of an internet of things-based prototype for smart home automation system," In *2018 International Conference on Recent Innovations in Electrical, Electronics & Communication Engineering (ICRIEECE)*, pp. 67-72, 2018. IEEE. [CrossRef] [Google Scholar] [Publisher Link]

[18] A. Jacobsson, M. Boldt, and B. Carlsson, "A risk analysis of a smart home automation system," *Future Generation Computer Systems*, vol. 56, pp.719-733, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[19] H. Singh, V. Pallagani, V. Khandelwal, and U. Venkanna, "IoT based smart home automation system using sensor node," In *2018 4th International Conference on Recent Advances in Information Technology (RAIT)*, pp. 1-5, 2018. IEEE. [CrossRef] [Google Scholar] [Publisher Link]

AUTHORS

J. Bharathi an academician with 26 + years of teaching experience. I obtained my MTech in Digital Systems and Computer Electronics from JNTU Hyderabad in the year 1997 and Ph.D. from JNTUA in 2016. My research interests lie in the domain of Signal and Image Processing, Pattern Recognition and Data Clustering.

C. Anuradha received her B.E. degree in Electrical and Electronics Engineering in Madras University and M.Tech degree in Power Electronics and Drives from SRM University, in 2004 and 2010 respectively. She has done her Ph.D. on Design and analysis of multi-port converter for micro grid applications in 2021 at SRMIST. She has published more than 21 journal papers on reputed and indexed journals in the field of Renewable energy system and power electronics converters design. She also filed and published 2 patents. She got 1 patent granted. Her research area of interests are Renewable energy sources, electric vehicles, energy management, design of power electronic converters etc. She has been working as an Assistant Professor in the Department of Electrical and Electronics Engineering, SRM Institute of Science and Technology, Kattankulathur since Feb 2007.She is a member of various professional societies like IEEE, ISCA, MISTE.

Arrived: 05.09.2024
Accepted: 01.10.2024

International Journal of System Design and Computing (IJSDC)
Volume 02, Issue 02, July - December (2024)

RESEARCH ARTICLE

DYNAMIC POWER ALLOCATION IN IOT-CLOUD ENVIRONMENT FOR HEALTHCARE APPLICATIONS

Kumarraja Andanapalli [1, *] and M. Suresh Kumar [2]

[1] Department of Electrical & Electronics Engineering, SRKR Engineering College, Bhimavaram, Chinamiram Rural, Andhra Pradesh 534204 India.

[2] Department of Computer Science and Engineering, Sri Venkateswara College of Engineering Sriperumbudur Chennai, 602117 India.

[*]*Corresponding e-mail: kumarraja@srkrec.ac.in*

Abstract – In the healthcare industry, the integration of the Internet of Things (IoT) and cloud computing (CC) enables access and sharing of worldwide health datasets. Thus, it solves complex problems like data security, privacy, and storage. However, the wide usage of cloud infrastructure increases traffic and reduces cloud performance. Hence, in this article, a novel hybrid Honey Pot-based Feed Feedback Neural System (HPbFFNS) framework was proposed to allocate resources optimally to medical applications (tasks). This framework incorporates the features of Honey pot optimization, and Feed Feedback Neural Network (FFNN). Initially, the health information of patients is collected using the IoT devices and forwarded into the gateway layer for further processing. The task scheduler in the gateway layer analyzes the resource availability, deadline, and priority of the incoming requests to reduce the response time, and waiting time. The honey pot fitness function in the resource allocator helps to allocate resources optimally to the tasks. Additionally, for verification purposes, the results are contrasted with those of current techniques. The experimental and comparative analysis confirms that the suggested model outperforms the traditional algorithms in terms of response time, energy consumption, and resource utilization.

Keywords – *Honey Pot Optimization, Feed Forward Neural System, Dynamic power allocation, Optimal Resource Allocation.*

1. INTRODUCTION

Cloud Computing (CC) is one of the developing techniques widely used in IoT-based applications because of its numerous advantages [1]. The basic principle behind CC is sharing of data/ information through the internet based on demand [2]. Cloud system contains hundreds of interconnected computers in a miscellaneous manner, where the data, files, and applications are accommodated [3]. CC incorporates parallel and distributed computing methods to provide sharing of resources like data/information, hardware, files, and software following the demand/ request on the cloud [4]. In the distributed system, CC offers a "Pay as you need" model [5]. Thus, the customer can access the computational software or platforms through the internet by paying the cost for the duration. Hence, the method eliminates the need to purchase of computational software or platforms for performing the task for the customers [6]. Moreover, the cloud system offers dynamic resource usage to the customers. In the cloud environment, virtual machines (VMs) are the processing units that share and compute the data/information dynamically as per the customer's demand/request [7]. Here, a huge number of VMs are interconnected and the resources are kept in a pre-emptive or non-pre-emptive manner resulting in the non-equal distribution of resources [8]. Thus, some VMs do not get the chance to share the resources. Moreover, when a task is assigned in the cloud, all VMs must implement the task faster to reduce computational time. In addition, all VMs must work in a parallel manner to minimize the time complexity [9]. This demands a proper scheduling algorithm to schedule the assigned task and complete the implementation within the available resources. When too many works are assigned to the VMs, they simultaneously work to complete the task [10].

While task assignation, the scheduler must check whether the VM is loaded or free. Moreover, it must assure that all the tasks are loaded to one VM leaving the other VMs extremely free or not utilized [11]. Thus, it is the responsibility of the scheduler to check whether the tasks are equally balanced in the cloud [12]. In recent times, one of the biggest challenges in CC is load balancing. To overcome this issue, an intelligent load-balancing mechanism is necessary for CC to safeguard the available resources [13]. Moreover, it improves the response time by implementing the assigned tasks faster [14]. Many researchers are conducted to develop an efficient load-balancing technique [15]. The major aim of the load-balancing strategy is to increase the response time of assigned tasks using the available resources [16]. The static load balancing method works properly when there are low fluctuations of load in VMs. Hence, it is not suitable for a cloud environment because the loads in the cloud vary unpredictably during the run time [17].

On the other hand, the dynamic load balancing strategy works properly at high fluctuating loads during the run time

[18]. Moreover, the high usage of the network and its resources demands an effective dynamic resource allocation (DRA) method [19]. Although various researchers were conducted to design an effective dynamic load balancing method, they face issues in time consumption, response time, and cost [20]. The existing techniques like deadline-based resource allocation algorithm [21], dynamic optimization mechanism [22], DRA approach based on particle swarm optimization (PSO) [23], etc., face challenges in cost and time complexity. Therefore, to overcome the existing techniques an optimized neural-based DRA system was designed in this article.

The key contribution of the presented work is defined below,

- Initially, the patient's health data is collected using the IoT devices in the IoT layer and forwarded into the IoT gateway.
- In the gateway layer, a hybrid HPbFFNS framework was designed to optimally allocate the resources in the IoT-cloud environment.
- The developed scheme estimates the deadline and priority of incoming requests to schedule the tasks optimally.
- The honey pot fitness function is integrated into the task allocator to determine the minimum resource availability in the cloud and to improve resource utilization.
- Finally, the performances of the presented model are evaluated and validated with existing techniques in terms of waiting time, response time, and energy consumption.

The presented article is sequenced as follows, the recent articles related to resource allocation (RA) are described in section 2, the problems of the existing techniques are illustrated in section 3, the proposed methodology is explained in section 4, the outcomes of the presented model are discussed in section 5, and the conclusion of the paper is mentioned in section 6.

2. RELATED WORKS

The following list includes a few recent studies on resource allocation.

In recent times, fog computing technology helps in supportive time-sensitive tenders related to the IoT. Conventionally, CC is widely used for processing IoT data. However, because of its high latency, it does not apply to time-sensitive applications. But RA in fog computing is a stimulating factor. Thus, Ranesh Kumar Naha et al. [21] presented a deadline-based RA algorithm to optimally allocate data to the users based on resource ranking. The developed scheme is employed in the CloudSim tool and the outcomes are determined. The performances of the developed scheme are validated in terms of processing time, delay in transmission, etc. However, the implementation cost is high in this method.

A fog computing network is one of the emerging techniques for the optimal sharing of information/ data in an IoT environment. However, allocating the resources optimally in the cloud environment is necessary to improve the response time. Thus, Zheng Chang et al. [22] presented a dynamic optimization mechanism for the CC network with multiple mobile devices to allocate resources optimally. This model utilizes hybrid radio and computational offloading method based on Lyapunov algorithm. Here, the main task is divided into several subtasks for reducing the complexity, and task load. Finally, the efficiency of the advanced model is authorized with experimental analysis.

The CC environment offers eliminates the need to purchase of computational software for performing specific tasks. But the high congestion that occurs in the network increases the latency, and cost. Hence, to overcome these issues D. Baburao et al. [23] developed a DRA approach based on PSO. This technique minimizes the task waiting time, latency, and power consumption, and increases the network's quality of service. Moreover, it provides better RA by eliminating the long-term inactive services from Radom Access Memory (RAM).

The incorporation of IoT technologies in industrial systems provides technical support. However, the heavy load congestion in the IoT-cloud environment reduces the response time and increases the waiting time. Therefore, Ying Chen et al. [24] suggested a load-balancing strategy based on the deep reinforcement technique. This developed model offers joint power control as well as DRA to mobile devices. Moreover, it minimizes the waiting time and latency in the network. Finally, the performances are estimated and evaluated by comparing them with traditional schemes.

Suchintan Mishra et al. [25] designed an effective RA system based on an analytic hierarchy process. The main aim of this method is to minimalize the latency acquired by each customer task. Generally, the cloud system is ineffective in analyzing latency-sensitive applications. This method employs a decision-making algorithm to provide load balancing in the cloud network dynamically. The experimental outcomes verify that the developed scheme outperforms the traditional methods. However, this RA system is not cost-efficient.

Nowadays, applications like business, mobile computing, and IT enterprises widely use CC platforms for storing and analyzing information. Hence, the resources such as CPU, input/output devices and memory can be used by the customers and charged as per the demand and usage. Therefore, J. Praveenchandar and A. Tamilarasi [26] presented an optimal power minimization method to enhance the effectiveness of RA. This utilizes a better task scheduling and prediction mechanism to offer optimal load balancing in the cloud network. This method overcomes the inability of the traditional scheme in offering optimal task scheduling and power consumption in CC. However, the latency parameter is not considered in this approach.

The cloud of things created by the incorporation of IoT and cloud environment increases the challenges in IoT and cloud areas. Hence, Seyedeh Maedeh Mirmohseni et al. [27] developed a load-balancing method based on the Markov model learning algorithm. In this model, the probability function is deployed to maximize network usage. The

presented approach is simulated in the cloudsim environment and the results are evaluated.

However, the data loads are high in the cloud because of its wide usage. Traditional RA fails to improve response time and resource utilization. Hence, Hongbin Liang *et al* [28] developed an intelligent resource management strategy using the artificial intelligence mode. This model utilizes adaptive and intelligent schemes to offer dynamic resource scheduling in the cloud network. Moreover, it employs self-learning, and reinforcement learning to reduce the waiting time which occurs because of the large processing of data.

Amir Javadpour *et al* [29] developed a RA system for peer-to-peer networks and IoT. This model utilizes a deep neural framework for training the system to offer dynamic load balancing. The deep neural-based technique can balance huge loads at high fluctuations. Moreover, an intelligent task scheduling algorithm was developed to check the availability of resources in the cloud, thereby reducing the waiting time. The experimental analysis of the presented algorithm shows that the performances like latency, waiting time, power consumption, etc., are minimized in this approach.

3. SYSTEM MODEL AND PROBLEM STATEMENT

IoT and CC infrastructure are integrated in smart healthcare to allow for real-time health monitoring. Sensors and IoT devices are used in IoT-based healthcare to gather patient data, such as blood pressure, cholesterol, body temperature, etc. These data can be used whenever needed because they have been moved to the cloud layer for storage. CC is the term used to describe the online distribution of computing services such as databases, software, storage, intelligence, etc. Allocating resources optimally is the main problem in CC. The RA in cloud infrastructure is displayed in Fig 1. Generally, in the cloud, the resources are allocated based on the user's request. However, the wide usage of the cloud network increases the load traffic, waiting time, latency, etc.

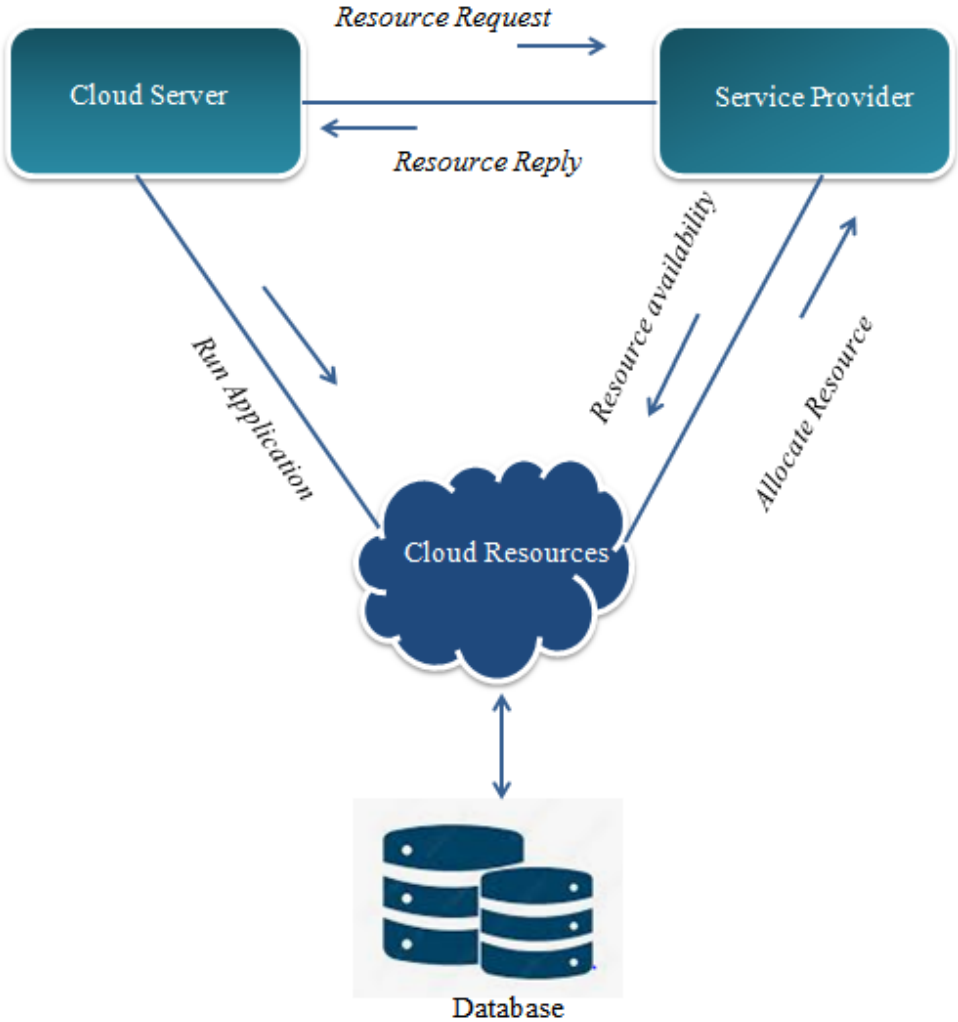


Figure 1. Resource Allocation in Cloud Environment

The RA in the cloud is of two major types namely: Dynamic and Static RA. But static RA is less effective because of high fluctuations in the cloud network. However, when compared with the static approach the dynamic method offers higher performance. Although different DRA methods are developed in the past, they face challenges in resource utilization, latency, and response time. Hence, an optimized neural-based RA method is established in this article to overcome the present challenges.

4. PROPOSED HPbFFNS FOR RESOURCE ALLOCATION

The use of medical sensors to gather patient health data from their locations is made possible by developments in smart healthcare. The cloud environment stores these IoT device data collection results for later processing. However, the heavy load traffic at the cloud servers causes failure in resource distribution, consumes more time, and increases the delay in the network. Hence, to resolve these issues in the IoT-cloud environment a novel hybrid Honey Pot-based Feed Forward Neural System (HPbFFNS) framework was developed in this paper to allocate resources to IoT healthcare applications in an optimal manner. This framework integrates the optimal features of Honey Pot Optimization [30], and Feed Forward Neural System [31]. In this framework, initially, the patient's medical data are collected using medical sensors in the IoT layer. These collected data from the IoT devices are forwarded to the IoT gateway layer for processing. In the gateway layer, the proposed HPbFFNS technique was incorporated to offer optimal load balancing.

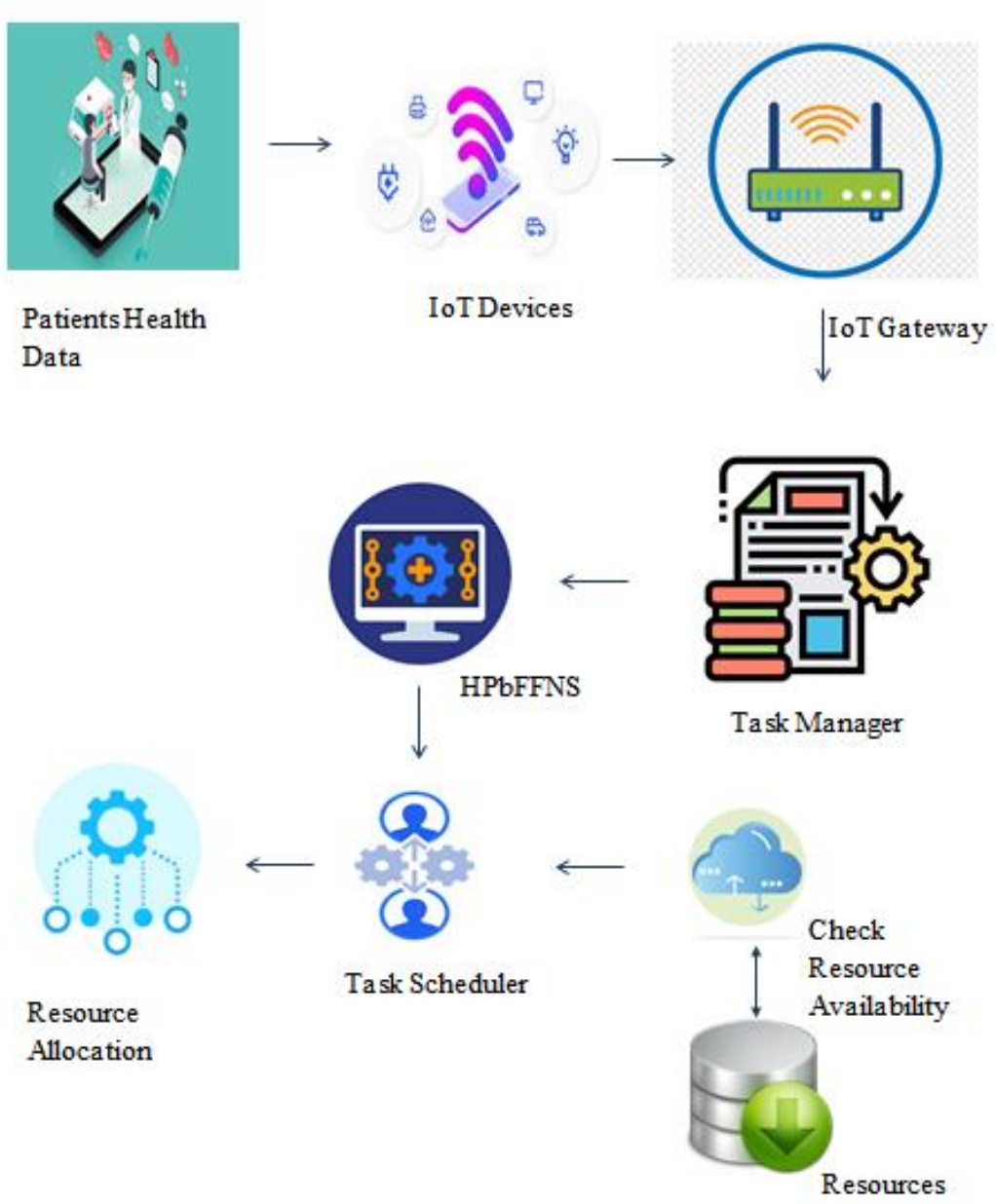


Figure 2. HPbFFNS Framework

In the gateway layer, there are three main components namely: Task manager, task schedular, and resource allocator. The task manager contains the view of all the processing tasks available in the network. It collects the task queues and forwards them to the scheduler. Moreover, it is the responsibility of the scheduler to check the availability of the resources in the cloud layer. Finally, the resource allocator assigns the resources as per the schedule to the healthcare applications considering the cost and energy consumption of each activity. The proposed framework is illustrated in Fig 2.

4.1. Data Collection

The integration of IoT in healthcare enables new scopes of patient healthcare through real-time monitoring and easy access to patients' health data. Moreover, it helps to facilitate healthcare progress virtually through telemedicine. The IoT devices utilized for gathering the patient's health information are a body temperature sensor, heart beat rate sensor, pulse-oximeter, fluid level sensor, etc. These sensors are placed inside the patient's body to provide real-time health monitoring. In addition, it eliminates the need for patients to travel to the healthcare units. In the cloud layer, the gathered information is stored for further processing. The IoT gateway is the hub that interconnects the IoT devices and sensors to the CC network. Moreover, it is responsible for filtering the collected data before transferring it out over the internet.

4.2. Task Scheduling

RA in an IoT-cloud environment involves three major steps namely: task management, RA and task scheduling. The task manager consists of the view of all incoming user requests (tasks) available in the cloud. It forwards the task queue to the task scheduler for RA. Here, the developed HPbFFNS model is integrated into the task schedular for reducing load traffic, waiting time, and latency. In the proposed model, the FFNN features are applied to the task scheduler to improve resource utilization. The FFNN is the simplest neural network, which can solve non-linear data faster. The problem-solving feature of FFNN is hybridized in the task scheduler to arrange the task effectively to improve the cloud performance. Initially, the scheduler checks the availability of the resources in the cloud. The task in the queue is expressed in Eqn. (1).

$$Q_s^{'} = [T_{s1}, T_{s2}, T_{s3}, T_{s4}, \ldots . T_{sm}] \quad (1)$$

Where, $Q_s^{'}$indicates the task queue, T_s represents the task, and m denotes the number of the task in the queue. Then, the scheduler estimates the deadline of each task based on the start time of a task, and the number of VMs required to complete the task. The deadline for an individual task is estimated using Eqn. (2).

$$Dl_T(T_s) = \sum_{i=0}^{m} Vm_n(Ex_T - St_T) \quad (2)$$

Here, Dl_T denotes the deadline of the task, Vm_n indicates the number of VMs required to complete the task, Ex_T represents the execution time of the task, andSt_T defines the start time of the task. Then, the scheduler analyzes the priority, and length of each task to schedule the tasks optimally. The priority and task scheduling are represented in Eqn. (3) and Eqn. (4).

$$P_r^{'}(T_s) = L_{gt}\left(\frac{Dl_T - Sl_T}{Vm_n}\right) \quad (3)$$

$$S_t^{'}[T_s] = \{P_r^{'}(T_{si}) \geq P_r^{'}(T_{sm})\} \quad (4)$$

Where$P_r^{'}$ indicates the task priority, L_{gt} denotes the length of the task, and $S_t^{'}$represents the task scheduling.

4.3. Resource Allocation

Resource allocation is the method of assigning the available resources in the cloud to the users over the internet based on their demand. After scheduling, the available resources from the cloud are distributed to each task optimally. In the proposed framework, the parameters like resource availability, resource demand, and task completion time. Initially, the minimum available resources in the cloud must be determined to reduce the waiting time. The minimum resource availability is expressed in Eqn. (5).

$$M_R^* = \frac{T_R - m_{ax}(T_s)}{m_{ax}(T_s) - m_{in}(T_s)} \quad (5)$$

Where M_R^*determines the minimum resource availability, $T_R$refers to the total available resources in the cloud, $m_{ax}(T_s)$and$m_{in}(T_s)$ defines the maximum and minimum RA to the incoming task. The honey pot fitness solution for RA is expressed in Eqn. (6).

$$R_{SA} = Hp_f + T_{set} \times (M_R^*) + D_{mr} \quad (6)$$

Where R_{SA}defines the RA function, Hp_findicates the honey pot fitness, D_{mr}denotes the resource demand, and T_{set}refers to the appropriate time duration of the task.

Figure 3. Flowchart of HPbFFNS

The honey pot fitness in the resource allocator enables it to analyze the resource demand and approximate time duration of the time. Thus, resource utilization is enhanced in the proposed method. The flowchart of the proposed model is illustrated in Fig 3.

5. RESULT AND DISCUSSION

A hybrid optimization-based RA framework was designed in this article to assign resources in the cloud environment for medical applications. Initially, the patient's health data was collected using IoT devices and sensors. The designed framework is implemented in MATLAB software, version R2020a.

5.1. Performance Analysis

To manifest the presented model performances, it is compared with existing techniques like Deadline-based Dynamic Resource Allocation (DbDRAA) [21], Improved Resource Allocation using Learning Classification Systems (IRA_LCS) [32], RA using Markov Learning Algorithm (RA_MLA) [27], and Task Scheduling based on Canonical PSO [33]. Moreover, the performance improvement percentage is also determined from the performance analysis.

5.1.1. Energy Consumption

Energy consumption defines the total energy consumed by the system to complete the task. For an efficient model, the energy consumption must be low to reduce the cost of the system. It is formulated in Eqn. (6).

$$E_{nc} = P_r \times \left(\frac{T_m}{1000}\right) \quad (6)$$

Where E_{nc}indicates the energy consumption, P_r denotes the applied power, andT_m refers to the time to complete the task.

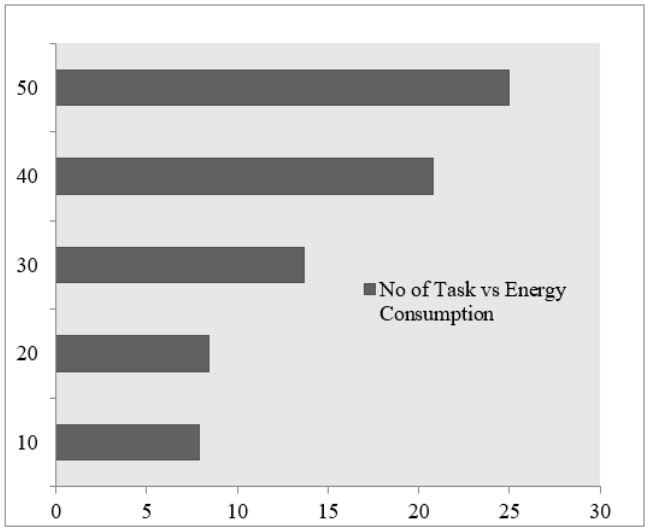


Figure 4. Energy Consumption Performance

In the presented work, the energy consumption is estimated by changing the number of tasks to 10, 20, 30, 40, and 50. When the task count increases, energy consumption increases in the system. It is observed that the energy consumption of the system for the assigned tasks is 7.89kWH, 8.4kWH, 13.7kWH, 20.8kWH, and 25kWH. The energy consumption performance analysis is shown in Fig 4.

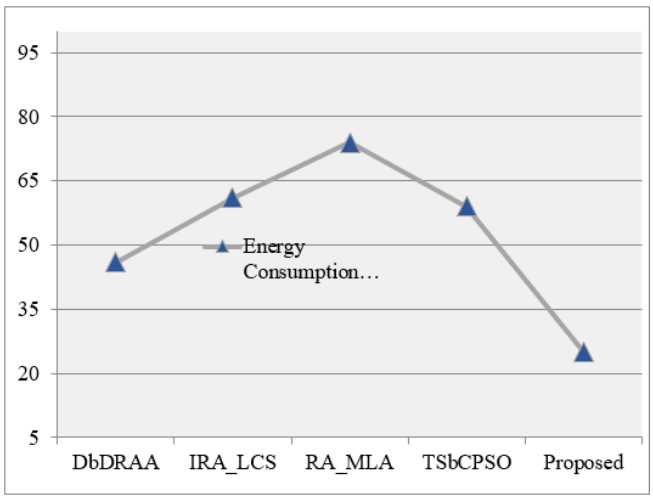


Figure 5. Comparison of Energy Consumption

Moreover, to manifest that the proposed model attained less energy to perform the task it is compared with some existing techniques like DbDRAA, IRA_LCS, RA_MLA, and TSbCPSO. The energy consumed by the traditional schemes is 46kWH, 61kWH, 74kWH, and 59kWH, respectively. This shows that the presented model consumed less energy to perform the tasks in the network. The energy consumption comparison is illustrated in Fig 5.

5.1.2. Response Time

Response time is the total time taken by the system to respond to the incoming request (task) for service. The service can be memory allocation, database query, etc. The response time is determined by adding the service and wait time to the network.

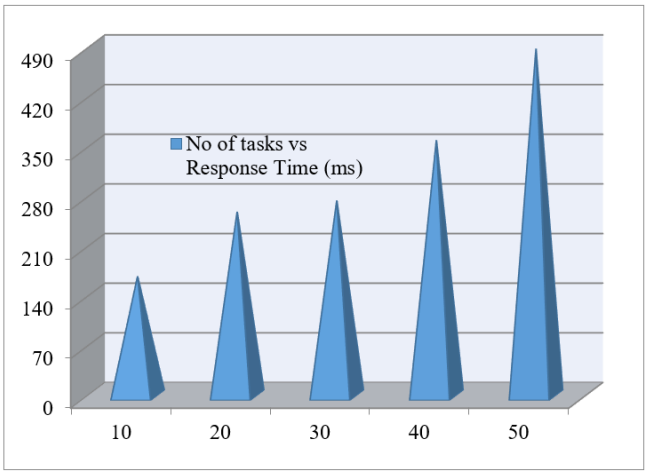


Figure 6. Response Time Analysis

Here, the response time of the presented system is determined by changing the number of tasks. The developed scheme achieved less response time of 168ms, 259ms, 275ms, 360ms, and 489ms, respectively for a different number of tasks (10, 20, 30, 40, and 50). The response time performance is illustrated in Fig 6.

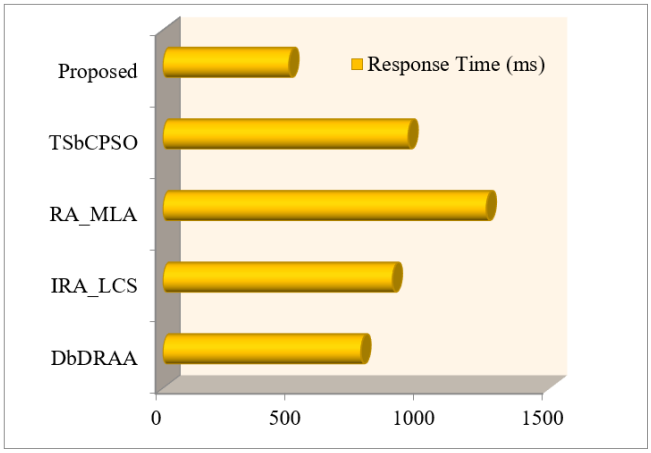


Figure 7. Comparison of Response Time

The response time attained by the existing algorithms is 768ms, 890ms, 1254ms, and 951ms. The response time comparison is shown in Fig 7. In addition, traditional frameworks like DbDRAA, IRA_LCS, RA_MLA, and TSbCPSO are executed on the same platform, and the response time is determined.

5.1.3. Resource Utilization

Resource utilization is one of the important parameters which determines cloud system performances. It defines the amount of resources used for the task. The presented algorithm earned a greater resource utilization rate of 96.7%. Further, it is compared with some existing techniques for validation purposes.

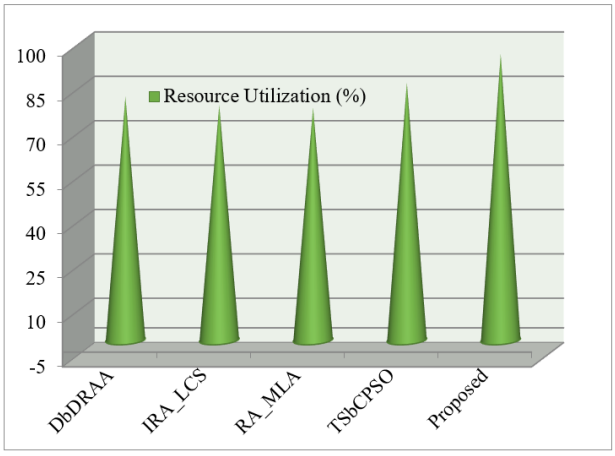


Figure 8. Comparison of Resource Utilization

The existing approaches like DbDRAA, IRA_LCS, RA_MLA, and TSbCPSO are executed in the same platform for medical applications. The resource utilization rate attained by the existing techniques is 82.3%, 79.46%, 78.6%, and 87%, respectively. The comparison of resource utilization rate is shown in Fig 8.

5.2. Discussion

In this article, an optimal dynamic resource allocation strategy was designed to allocate resources for IoT medical applications in the cloud infrastructure. It combines the honey pot optimization and feedback neural system to schedule and allocate resources optimally. The comparative performance of the developed scheme is tabulated in Table 1.

Table 1. Comparative Analysis

Techniques	*Response Time (ms)*	*Energy Consumption (kWH)*	*Resource Utilization (%)*
DbDRAA	768	46	82.3
IRA_LCS	890	61	79.46
RA_MLA	1254	74	78.6
TSbCPSO	951	59	87
Proposed	489	25	96.7

6. CONCLUSION

In recent times, one of the major concerns in the IoT-cloud infrastructure is the optimal resource allocation to incoming requests from different clients. To resolve this issue, an optimized neural-based DRA strategy was proposed in this article. The developed scheme involves three major steps namely: data collection, task scheduling, and RA. The FFNN attributes and honey pot fitness are integrated into the developed scheme to improve resource utilization by optimally assigning the available resources in the cloud. Furthermore, the performances of the accessible algorithm were estimated and validated with a comparative assessment. In addition, the performance improvement rate is determined from the comparative analysis. In the presented approach, the resource utilization is enhanced by 9.3%, the response time is minimized by 279ms, and the energy consumption is reduced by 21kWH. Thus, the designed model allocates resources optimally to the medical applications and improves the performances.

CONFLICTS OF INTEREST

Not applicable.

FUNDING STATEMENT

Not applicable.

ACKNOWLEDGEMENTS

The author would like to express his heartfelt gratitude to the supervisor for his guidance and unwavering support during this research for his guidance and support.

REFERENCES

[1] S. B. Sangeetha, R. Sabitha, B. Dhiyanesh, G. Kiruthiga, N. Yuvaraj, and R. A. Raja, "Resource management framework using deep neural networks in multi-cloud environment," Operationalizing Multi-Cloud Environments: Technologies, Tools and Use Cases, pp. 89-104, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[2] M. Kandan, A. Krishnamurthy, S. A. M. Selvi, M. Y. Sikkandar, M. A. Aboamer, and T. Tamilvizhi, "Quasi oppositional Aquila optimizer-based task scheduling approach in an IoT enabled cloud environment," The Journal of Supercomputing, vol. 78, no. 7, pp. 10176-10190, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[3] K. Peng, H. Huang, B. Zhao, A. Jolfaei, X. Xu, and M. Bilal, "Intelligent computation offloading and resource allocation in IIoT with end-edge-cloud computing using NSGA-III," IEEE Transactions on Network Science and Engineering, vol, 10, no. 5, pp. 3032-3046, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[4] R. Jeyaraj, A. Balasubramaniam, A. K. MA, N. Guizani, and A. Paul, "Resource management in cloud and cloud-influenced technologies for internet of things applications," ACM Computing Surveys, vol. 55, no. 12, 1-37, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[5] V. Jain, and B. Kumar, "Auction based cost-efficient resource allocation by utilizing blockchain in fog computing," Transactions on Emerging Telecommunications Technologies, vol. 33, no. 7, pp. e4469, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[6] Q. Li, P. Kumar, and M. Alazab, "IoT-assisted physical education training network virtualization and resource management using a deep reinforcement learning system," Complex & Intelligent Systems, pp. 1-14, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[7] Z. Wang, T. Lv, and Z. Chang, "Computation offloading and resource allocation based on distributed deep learning and software defined mobile edge computing," Computer Networks, vol. 205, pp. 108732, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[8] Z. Xiang, Y. Zheng, D. Wang, M. He, C. Zhang, and Z. Zheng, "Robust and cost-effective resource allocation for complex iot applications in edge-cloud collaboration," Mobile Networks and Applications, vol. 27, no. 4, pp. 1506-1519, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[9] Y. Kumar, S. Kaul, and Y. C. Hu, "Machine learning for energy-resource allocation, workflow scheduling and live migration in cloud computing: State-of-the-art survey," Sustainable Computing: Informatics and Systems, vol. 36, pp. 100780, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[10] N. Nithiyanandam, M. Rajesh, R. Sitharthan, D. Shanmuga Sundar, K. Vengatesan, and K. Madurakavi, "Optimization of performance and scalability measures across cloud based IoT applications with efficient scheduling approach," International Journal of Wireless Information Networks, vol. 29, no. 4, pp. 442-453, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[11] S. Masoudi, and F. Safi-Esfahani, "SM@ RMFFOG: sensor mining at resource management framework of fog computing," The Journal of Supercomputing, vol. 78, no. 17, pp. 19188-19227. 2022. [CrossRef] [Google Scholar] [Publisher Link]

[12] H. Chen, W. Qin, and L. Wang, Task partitioning and offloading in IoT cloud-edge collaborative computing framework: a survey. Journal of Cloud Computing, vol. 11, no. 1, pp. 86, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[13] S. Abedi, M. Ghobaei-Arani, E. Khorami, and M. Mojarad, "Dynamic resource allocation using improved firefly optimization algorithm in cloud environment," Applied Artificial Intelligence, vol. 36, no. 1, pp. 2055394, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[14] R. Rani, V. Kashyap, and M. Khurana, "Role of IoT-cloud ecosystem in smart cities: review and challenges," Materials Today: Proceedings, vol. 49, pp. 2994-2998, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[15] F. Firouzi, B. Farahani, and A. Marinšek, "The convergence and interplay of edge, fog, and cloud in the AI-driven Internet of Things (IoT)," Information Systems, vol. 107, pp. 101840, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[16] M. Prakash, V. Vijayaganth, F. D. Shadrach, R. Menaha, T. Daniya, and T. Guha, Improved Political Optimizer and Deep Neural Network-based Resource Management Strategy for fog Enabled Cloud Computing. In 2022 IEEE 2nd Mysore Sub Section International Conference (MysuruCon), pp. 1-6, 2022. IEEE. [CrossRef] [Google Scholar] [Publisher Link]

[17] S. Rahul, and V. Bhardwaj, "Optimization of Resource Scheduling and Allocation Algorithms," In 2022 Second International Conference on Interdisciplinary Cyber Physical Systems (ICPS), pp. 141-145, 2022. IEEE. [CrossRef] [Google Scholar] [Publisher Link]

[18] G. Vijayasekaran, and M. Duraipandian, "An efficient clustering and deep learning based resource scheduling for edge computing to integrate cloud-IoT," Wireless Personal Communications, vol. 124, no. 3, pp. 2029-2044, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[19] X. Sun, J. Jia, Z. Liu, Y. Li, B. Sun, and D. Liu, Resource Allocation and Load Balancing Based on Edge Computing in Industrial Networks. In 2022 IEEE International Conference on Smart Internet of Things (SmartIoT), pp. 250-251, 2022. IEEE. [CrossRef] [Google Scholar] [Publisher Link]

[20] R. Pradhan, A. K. Dash, and B. Jena, Resource management challenges in IoT based healthcare system. Smart Healthcare Analytics: State of the Art, pp. 31-41. 2022. [CrossRef] [Google Scholar] [Publisher Link]

[21] R. K. Naha, S. Garg, A. Chan, and S. K. Battula, "Deadline-based dynamic resource allocation and provisioning algorithms in fog-cloud environment," Future Generation Computer Systems, vol. 104, pp. 131-141, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[22] Z. Chang, L. Liu, X. Guo, and Q. Sheng, "Dynamic resource allocation and computation offloading for IoT fog computing system," IEEE Transactions on Industrial Informatics, vol. 17, no. 5, pp. 3348-3357, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[23] D. Baburao, T. Pavankumar, and C. S. R. Prabhu, "Load balancing in the fog nodes using particle swarm optimization-based enhanced dynamic resource allocation method," Applied Nanoscience, 13(2), 1045-1054, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[24] Y. Chen, Z. Liu, Y. Z hang, Y. Wu, X. Chen, and L. Zhao, "Deep reinforcement learning-based dynamic resource management for mobile edge computing in industrial internet of things," IEEE Transactions on Industrial Informatics, vol. 17, no. 7, pp. 4925-4934, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[25] S. Mishra, M. N. Sahoo, S. Bakshi, and J. J. Rodrigues, "Dynamic resource allocation in fog-cloud hybrid systems using multicriteria AHP techniques," IEEE Internet of Things Journal, vol. 7, no. 9, pp. 8993-9000, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[26] J. Praveenchandar, and A. Tamilarasi, "Retracted article: dynamic resource allocation with optimized task scheduling and improved power management in cloud computing," Journal of Ambient Intelligence and Humanized Computing, vol. 12, no. 3, pp. 4147-4159, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[27] S. M. Mirmohseni, C. Tang, and A. Javadpour, "Using Markov learning utilization model for resource allocation in cloud of thing network," Wireless Personal Communications, vol. 115, pp. 653-677, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[28] H. Liang, X. Zhang, X. Hong, Z. Zhang, M. Li, G. Hu, and F. Hou, "Reinforcement learning enabled dynamic resource allocation in the internet of vehicles," IEEE Transactions on

Industrial Informatics, vol. 17, no. 7, pp. 4957-4967, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[29] W. Wei, R. Yang, H. Gu, W. Zhao, C. Chen, and S. Wan, "Multi-objective optimization for resource allocation in vehicular cloud computing networks," IEEE Transactions on Intelligent Transportation Systems, vol. 23, no. 12, pp. 25536-25545, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[30] Á. Balogh, M. Érsok, L. Erdődi, A. Szarvák, E. Kail, and A. Bánáti, "Honeypot optimization based on CTF game," In 2022 IEEE 20th Jubilee World Symposium on Applied Machine Intelligence and Informatics (SAMI), pp. 000153-000158, 2022. IEEE. [CrossRef] [Google Scholar] [Publisher Link]

[31] J. Guo, C. K. Wen, and S. Jin, "Deep learning-based CSI feedback for beamforming in single-and multi-cell massive MIMO systems," IEEE Journal on Selected Areas in Communications, vol. 39, no. 7, pp.1872-1884, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[32] M. Buvana, K. Loheswaran, K. Madhavi, S. Ponnusamy, A. Behura, and R. Jayavadivel, "Improved Resource management and utilization based on a fog-cloud computing system with IoT incorporated with Classifier systems," Microprocessors and Microsystems, 103815, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[33] M. Z. Hasan, and H. Al -Rizzo, Task scheduling in Internet of Things cloud environment using a robust particle swarm optimization. Concurrency and Computation: Practice and Experience, vol. 32, no. 2, pp. e5442, 2020. [CrossRef] [Google Scholar] [Publisher Link]

AUTHORS

Kumarraja Andanapalli received his Ph.D. degree in Electrical Engineering from NIT Raipur in the year 2023. Received M.E & B.E Degree from SRKR Engineering College, Bhimavaram and ANITS, visakhapatnam from India in 2013 & 2008. Presently, working as an Assistant Professor in the EEE department in SRKR Engineering College, AP. His area of interest includes power system protection, Artificial Intelligence, Signal Processing Techniques, Internet of things.

Suresh Kumar M currently works as Associate Professor in the department of Computer Science and Engineering at Sri Venkateswara College of Engineering and has 17 years of Teaching experience. Suresh Kumar M completed his UG in the department of Information Technology at paavai engineering college, Namakkal and also completed his PG in the department of CSE with the specialisation of System Engineering and operational Research at Anna University in 2010. Also, I completed my doctorate in 2024 in Wsn. My primary research areas are data mining, cyber security, Deep learning and WSN. Also, I am working as an Assistant Placement officer in my college.

Arrived: 12.09.2024
Accepted: 15.10.2024

International Journal of System Design and Computing (IJSDC)
Volume 02, Issue 02, July - December (2024)

RESEARCH ARTICLE

NETWORK INTRUSION DETECTION SYSTEM WITH AN EDGE BASED HYBRID FEATURE SELECTION APPROACH

A. Biju [1,*], and S. Wilfred Franklin [2]

[1] Department of Computer Science and Engineering, Maria College of Engineering and Technology, Attoor, Thiruvattaru, Tamil Nadu 629177 India

[2] Department of Electronics and Communication Engineering, C.S.I Institute of Technology, Thovalai, Kanyakumari, Tamil Nadu 629302 India.

[*]*Corresponding e-mail: bijubijua@gmail.com*

Abstract – An intrusion detection system looks through network data to find both legitimate and malicious activity. This study can detect new attacks, which is especially useful in IoT situations. Deep Learning (DL) has demonstrated its superiority in solving challenging real-world issues such as NIDS. This method, however, necessitates more processing resources and takes a lengthy time. During a classification process, feature selection is critical in selecting the best attributes that best describe the goal concept. A novel Network intrusion detection (NEST) technique has been proposed to develop an improved edge-based hybrid feature selection approach, which is a deep learning method for detecting malicious traffic. The Enhanced BPSO technique overcomes the difficulty of BPSO feature selection by combining Binary Particle Swarm Optimization (BPSO) and correlation– based (CFS) traditional statistical feature selection. Three intrusion detection module having three classifiers make up the proposed system. The Signature Detection Module (SDM) examines threats and classifies it as unknown, normal, or intruder based on matching signatures utilizing the Generalized Suffix Tree (GST) algorithm. The Anomaly Detection Module (ADM) employs deep Q-learning to detect unknown attacks. The Hybrid Detection Module (HDM) employs the Meta-AdaboostM1 algorithm. Results of simulation show that the proposed protocol increases Detection Rate, computation time, and False Alarm Rate when compared to the existing XGBOOST-DNN, HNGEA, and HDLNIDS methods. The detection rate of the proposed NEST method is 4.55%, 6.47%, and 10.32% higher than the existing XGBOOST-DNN, HNGEA, and HDLNIDS, techniques respectively.

Keywords *– Network Intrusion Detection System, Deep Learning, Generalized Suffix Tree, Q-learning, Meta-AdaboostM1.*

1. INTRODUCTION

Network intrusion detection systems (NIDS) have really been successfully advanced in academia and business in response to the increasing number of cyber-attack on government and commercial enterprises around the world [1]. The annual cost of cybercrime is gradually rising. The most dangerous cyber-crimes include malicious insiders, dos attack, and web- based attacks [2]. If harmful malware infiltrates a system, it may result in the loss of intellectual property, as well as the disruption of a country's critical national infrastructure [3]. Businesses utilise antivirus software, and IDS to safeguard computer systems from unauthorized access (NIDS) [4].

As a product of the expanding quantity of data and the increased need edge computing is becoming popular for processing data a critical crossroads in history [5]. Edge computing increases service stability and delivers artificial intelligence services for terminal devices and data that are rapidly developing Smart terminals and other edge computing devices are near to the source of data [6]. It responsible for processing data at the network's edge. Near-end service. and also, proximity and location awareness, can benefit users [7]. It is fast, better real-time, and much more secure in terms of information processing [8]. It could also aid in the reduction of expenditures and alleviation of network bandwidth congestion caused by cloud computing's high energy usage. Manufacturing, energy, smart homes, and transportation are just a few of the industries that use edge computing [9,10].

Any device, software, or hardware capable of detecting suspicious behaviour or preset threats and then implementing appropriate countermeasures will be considered an intrusion detection system (IDS) [11]. These devices have developed into crucial tools for detecting and protecting a network in the previous stages. Every day, new intrusions are detected, causing increasing amounts of damage and potentially affecting a company's information system's operation [12]. Today's packet filtering systems look for signals of malicious activity or unauthorized access in packets sent over the network. In packets transferred over the network, IDS systems seek for evidence of malicious activity or unauthorized and undetected access [13]. As computer attacks get more sophisticated and identification of breach has become more complex, this problem has emerged as one of the most important challenges in the world of computer

security. To overcome these issues, a novel Network intrusion detection (NEST) technique has been proposed. The following are the primary contributions.

- Initially the collected data are preprocessed using data conversion and data normalization technique.
- The study introduces an improved edge-based hybrid feature selection approach that combines BPSO and CFS for the improved selection of relevant features for detecting malicious traffic, optimizing the trade-off between processing resources and classification accuracy.
- The IDS is structured into three distinct modules: SDM, ADM, and HDM. Each module specializes in different aspects of intrusion detection, such as signature matching, anomaly detection using deep Q-learning, and hybrid classification using Meta-AdaBoostM1, respectively.
- Extensive simulations using the NSL-KDD dataset authorize the effectiveness of the proposed method.

The explanation that follows concerns the next half of this study: The literature is consulted in Section II to assess the research. In Section III, the proposed system is explained in full. The conclusion is found in Section V, whereas the result and discussion are found in Section IV.

2. LITERATURE SURVEY

In 2019, Papamartzivanos, et al. [14] suggested an innovative approach that delivers a climbable, self-adaptive, and independent abuse by uniting the advantages of self-taught erudition with MAPE-K frameworks. IDS. The experimental findings show that our approach can revitalize the IDS and eliminate the requirement for manual training set refreshes. The suggested approach is assessed using a number of classification metrics, and the results show that under crucial circumstances where a statically trained IDS is rendered useless, the ADR of the IDS can rise to 73.37%.

In 2020, Devan et al., [15] suggested XGBoost–DNN method uses a deep neural network to classify network invasions after applying the XGBoost algorithm for feature selection. The suggested framework is authorized by cross-validation, and its performance is related with well-known shallow ML methods such as SVM, naïve, and Bayes logistic regression. A deep learning model consistently outperforms prior models in terms of classification accuracy, as evidenced by the reported findings.

In 2020, Venkatraman, and B. Surendiran [16] suggested adaptive hybrid IDS using a controller technique for timed automata. The experimental findings demonstrate the suitability of suggested method, for smart city applications. It also demonstrates its accuracy (99.06%) in identifying various types of attacks in IoT environments, including replay, zero-day, and DoS attacks.

In 2020, Elhefnawy, et al. [17] proposed a framework for Hybrid Nested Genetic-Fuzzy Algorithms (HNGFA) to provide security experts with highly optimized outputs for the classification of major and small danger categories. The findings demonstrate that, in various configurations on complicated datasets, the HNGFA performs better than alternative methods in terms of detection, investigation, and dynamic regulations for all minor attack types with excellent precision.

In 2021, Seo, and Pak, [18] suggested a two-level intrusion detection method that has a high finding accuracy. The level 1 classifier first extracts a limited set of features from the packet to facilitate rapid organization and real-time threat detection. Since the level 2 classifier only handles flows that the level 1 classifier was unable to classify, the traffic is manageable by a labor-intensive machine learning-based classifier.

In 2023, Qazi, et al. [19] suggested a system HDLNIDS for identifying network intrusions. Experiments are conducted assessing the suggested method's efficacy using publicly available benchmark CICIDS-2018 data. The study's conclusions show that with an average accuracy of 98.90%, HDLNIDS performs better than existing intrusion detection techniques at identifying malicious attempts.

In 2023, Hnamte, et al. [20] suggested a cutting-edge, two-stage deep learning method for attack detection that combines AE with LSTM. It can analyze network activities and makes use of a very effective framework. The desired LSTM-AE's ideal network parameters are found using the CICIDS2017 and CSE-CICDIS2018 datasets. The outcomes of the experiments prove the efficiency of the suggested hybrid model and its applicability in identifying attacks in contemporary settings.

3. NETWORK INTRUSION DETECTION (NEST)

In this section, a novel Network intrusion detection (NEST) technique has been proposed for effective intrusion detection in the network environment. Initially methods for data conversion and data normalization are used to preprocess the gathered data. In order to optimize the trade-off between processing resources and classification accuracy, the paper presents an enhanced edge-based hybrid feature selection strategy that combines BPSO and CFS for the improved selection of relevant features for identifying hostile traffic. SDM, ADM, and HDM are the three main modules that make up the IDS. Aspects of intrusion detection that each module focuses on differently include signature matching, hybrid classification using Meta-AdaBoostM1, and anomaly detection using deep Q-learning. The efficacy of the suggested strategy is authorized by extensive simulations with the NSL-KDD dataset. Figure 1 shows the framework of proposed methodology.

3.1. Data Collection

The dataset is exclusively responsible for testing, inspecting, and evaluating the way the discovery scheme behaves, and it plays a critical part in obtaining a better outcome. A high-performance one could deliver beneficial results not only for an offline device, but also in a real-world setting. The majority of the writers used the NSL-KDD datasets, which are a better version of the KDD CUP 99 database that eliminates duplicate data and selects articles based on their proportion. During pre-processing, it contains

148,517 document each with 41 attribute and a class mark. The five types are DoS, U2R, probing, R2L, and normal.

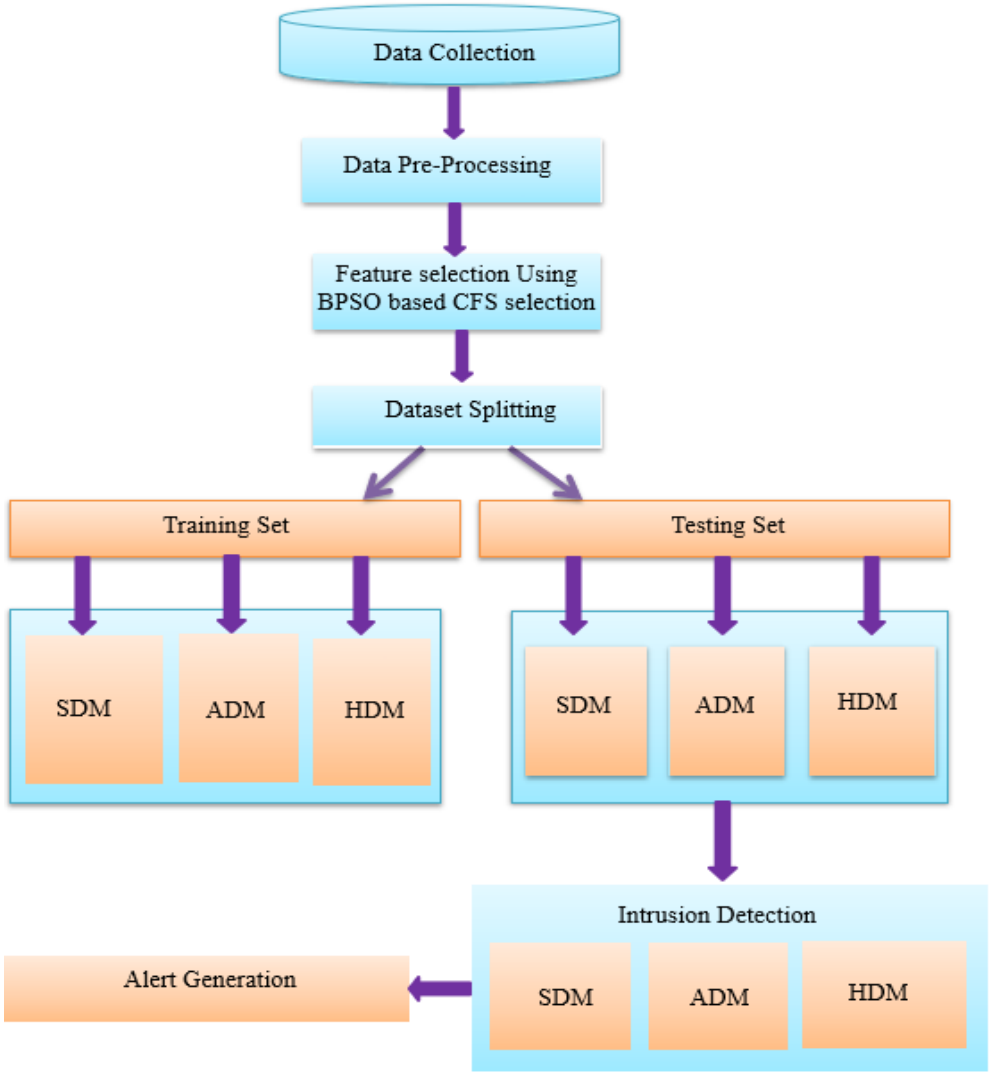


Figure 1. Architecture of proposed NEST methodology

3.2. Data Pre-Processing

The two primary methodologies for data pre-processing are data conversion and data normalisation. Data conversion converts components of traffic from nominal to numeric to guarantee that almost all data has numerical for processing by intrusion-detection system. Data normalisation is used to condense a vast range of values into a manageable range. Furthermore, during normalisation, null values are deleted. To normalise high numbers and decrease their significance, we utilise a minimum–maximum scaling method to put values between 0 and one, as shown by the equation (1)

$$f_{ij} = \frac{f_{ij} - min(f_{ij})}{max(f_{ij}) - min(f_{ij})} \quad (1)$$

where , f_{ij} is the value of the feature in the dataset matrix's row i and column j.

3.3. Feature selection and ranking

For the purpose of feature selection, the cross-validation technique was utilised to see how well the statistical analysis results might be generalised to an in-dependent dataset. The Enhanced BPSO approach is used to focus on feature selection using swarm intelligence (BPSO). Because of the obstacles posed by Big Data, feature selection in intrusion detection improves classification performance by minimising computational operations. When opposed to normal KDD data sets for Intrusion Detection, feature selection plays a key role with great quality real - world data sets due to the selection of only the most connected features with defined classes. The collected data required to be separated into training and testing data in order to analyse those produced models.

3.3.1. Correlation–Based Feature Selection

CFS is a filters method whose main goal is to discover the optimal solution in search space by evaluating the relevancy of an extracted features to a class and the redundancy between selected subset of features. The features are chosen based on the correlation function's feature subset assessment result. This indicates that selected characteristics are most closely related to the class, and not to one another.

As shown in Equation (2) every feature with only a highest score predicts classes in the subspace as well as other characteristics.

$$CS = \frac{fdef}{\sqrt{f+f(f-1)d_{ff}}} \quad (2)$$

Where CS is the score for a s extracted features with f features, dcf is the mean level of similarity among features and the class label, and dff is the degree of inter-correlation mean between qualities, d_{ff} is the degree of inter-correlation mean with both features, and d_{ff} is the degree of inter-correlation mean between features, and d_{ff} is the degree of inter-correlation mean between features, and d_{ff}. A correlation approach known as feature subsets is used to assess CFS. Larger dcf or smaller d_{ff} result in greater evaluation value in specific subsets.

3.3.2. Binary Particle Swarm Optimization (BPSO)

BPSO is a variant of the PSO algorithm adapted for binary search spaces. It's particularly useful for feature selection in ML, where the goal is to identify a subset of features that leads to the best performance of a model. As illustrated in Equations (3) & (4), each particle in PSO adjusts its rate of change and location in every iteration depending on personal experiences (pbest) and the swarm's greatest experience (gbest) (3). The performance of all particles is measured using defined cost functions at the end of every iteration.

$$V_j[st+1] = W \times V_j[st] + F1d1\big(P_{jbest}[st] - P_j[st]\big) + F2d2\big(G_{best}[st] - P_j[st]\big) \quad (3)$$

$$P_j[st+1] = P_j[st] + V_j[st+1] \quad (4)$$

Each particle j is iterated at each iteration. Obtain three vectors of length N that represent the problem dimension: velocity, location, and personal best. The end condition is accomplished when the enhanced size of the global finest is less than the stop value () or when the maximum iteration count is reached, and PSO stops.

The number of populations to implement the BPSO is set at 100, the number of iterations is set at 10. Initialize swarm at random, with $X = (x1, x2, \ldots, xn)$ representing a particle as a feature vector and y [0,1] signifying a class label, with 0,1 and 1, respectively, corresponding to normal and abnormal. Then add the following value in the variable:

W stands for the inertia weight, which regulates the particle's velocity impact on present iteration and is normally between [0.4,0.9]. The acceleration coefficients F 1 and F 2 are constants with a range of [0.5]. while d 1 and d 2 are random counts in the range [0,1]. On the velocity changes, these parameters scale both personal and swarm knowledge. As a result, as stated in Equation (5), utilise the Activation Function to estimate each particle's fitness value, and then choose the particle with the best value.

$$F(X) = \alpha(1 - Pr) + (1 - a)\left(1 - \frac{N_s}{N_v}\right) \quad (5)$$

N_s is the size of the feature subset that was tested, while N_s denotes the total number of input variables available. Pr is a metric measuring how well a classifier performs. Total accuracy is represented on the left side of the equation, while the proportion of utilised features is represented on the right. Regular PSO equations are converted to operate in binary space to create BPSO. In addition, the sigmoid function in Equation (6) was utilised to convert $V(st+1)$ to the [0,1] range. In BPSO, the velocity vector represents the probability of a component in the position vector taking value 1, which is given in equation (7).

$$S(V\,j^{st\cdot 1}) = \frac{1}{1+e^{-(Vj^{st-1})}} \quad (6)$$

$$Pj^{st+1} = \begin{cases} 1 & if\ rand() < S(Vj^{st+1}) \\ 0 & otherwise \end{cases} \quad (7)$$

where $rand$ () is a value chosen at random from the range [0,1]. PSO then generates an optimum solution, which is the best vector globally, and checks for the stop condition, which must be met for PSO to exit.

i. Enhanced BPSO features based on CFS selection

CFS classical statistical approach has been developed to progress the feature selection of standard BPSO Algorithm. Algorithm 1 shows the pseudocode for improved BPSO algorithm. The following approach is used to implement the Enhanced BPSO Algorithm:

1. Using a CFS-based correlation equation, calculate a score for each attribute.
2. Set a threshold and then choose every feature that are greater than it.
3. Run BPSO on a subset of features you've chosen.
4. Feature selection to eliminate features and select the best group of features

Algorithm 1 The Improved BPSO Algorithm

Input: Training and testing sets;

output: Xbest subset of characteristics begin

initialize N particle population Xj best =(X1,...,DT), j=1,2,3,…..,N

set up fit(Xj) and Xbest.

While j=max number of iterations,

initialise solution storage, use Fit temp (j) and X temp (j). update Xj and Vj

for $\boldsymbol{j = 0\ to\ n}$.

if rand () returns Vj select Xj from X.

if Calculate Fit returns true create a new X new

end (X new)

temper (j) = temper (X new)

If Fit ($X\ j$) equals Fit (X new), then (X new) If Fit (X new)>=max Fit temp,

then X best = X new

end if end

3.4. Signature Detection Module (SDM)

SDM is used to identify all attack patterns by matching signatures maintained in the system. Position Aware Distribution Signature (PADS) is used to construct the signatures. A Generic Suffix Tree (GST) is used to keep the signature in the repositories, that might check signature in a time that is asymptotically ideal. The signature repository is checked using $O(m+n)$ if the signatures are m bytes long. In the SDM, the Light-Net technique is employed to detect network attack. Continuous weight networks make up the LightNet. The most of the weights are zero, and those that aren't can only be -1 or +1. Light Net uses synaptic pruning training to construct the active function, which is represent by odd or hyperbolic tangent expression. As an outcome, the arbitrary location is specifically defined in equation (8)

$$\mathrm{AT} = \tanh(x-p) + \tanh\left(-(x-p) + \sigma\right) \quad (8)$$

There are three layers to Light Net: Hidden, input, and output. Packet feature is considering input, and the hidden layer HMS is also clustering related packet feature. SDM in the HDM detects threats by comparing signature in the tree. Intruder packets are reported, and an ADM examines anomalous packets to determine the sort of assault.

3.5. Anomaly detection Module (ADM)

A deep Q-learning algorithm analyses SNR and bandwidth characteristics classifies assaults as DoS, U2R , or (R2L) in the ADM. The agent learns about its surrounds by Q-learning, which results in a Q-table matrix containing states and action. In contrast, Q-learning is best suited to small-scale situations, while the Internet of Things is a vast system. As a result, Q-learning and deep learning are combined to build a Deep Q-learning system capable of simultaneously processing several unexpected attack packets. Each packet is an input to the input layer of the Deep Q-learning algorithm, containing bandwidth and SNR. The best classification technique is deep learning, which works well with big data sets. Because of the grouping of these characteristics, deep Q-learning can apply categorization. Let's call the states ($S1$, $S2$, $S3$, … St) and ($A1$, $A2$, $A3$, … At) correspondingly. In Deep Q-learning, the Q-value is calculated using the following given equation (9)

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha[R_{t+1} + \gamma maxQ(S_{t+1}, a) - Q(S_t, A_t) \quad (9)$$

According to the attack detection decision, R_{t+1} is specified as a reward by 1 on each timestep. Deep Q- learning uses an epsilon-greedy policy to carry out the activities. The suggested approach uses the packet's SNR and bandwidth, as well as other critical packet properties, to detect four potential assaults (DoS, Probe, U2R, and R2L) that aren't described in SDM. Algorithm 2 shows the pseudocode for Q-learning algorithm.

Algorithm 2. Q-learning

Begin

For all (St, At);

If S is terminal then

Compute new initial state from the reward Else

S← S;

Return attack type{DoS,Probe,U2R,R2L} End

The traffic's byte frequency distribution is determined and compared to the regular traffic distribution. A huge discrepancy is seen as unusual. The signature length w is positioned in relation to the byte frequency and percentage in an anomalous signature, where W is the signature width in bytes. It is categorised as Dos, Probe, U2R, R2L, or normal using ADM.

3.6. Hybrid Detection Module (HDM)

The categorization technique in the previous modules (SDM and ADM) was shown to have significant problems. When anticipating unexpected sample, the SDM has been unable to recognize unexpected classes, but the ADM has a higher probability of false alarms. As a result, the Hybrid technique was employed to achieve a balance between detection precision and false alarm rate. This method involves training several classes on the same dataset and then blending the results. It is thought that the Meta-AdaBoostM1 approach improves detection precision. AdaBoost is a boosting method for creating new classifier that look for and focus on examples that were previously misinterpreted by a classifier. Typically, this method uses the training data to train a weak classifier. It uses a decision stump for the weak classifier. A decision tree with only one level is called a decision stump. One internal node (root) connects all of the terminal nodes together. Using the same training dataset, the weak classifier is retrained with modified weights for precise classification. Reclassifying the weak classifier is done using a strong classifier. One powerful classifier that is employed is the meta-AdaboostM1 method. One powerful classifier that is employed is the meta-AdaboostM1 method. Algorithm 3 shows the pseudocode for Q-learning algorithm.

Algorithm 3 HDM Meta adaBoostM1 Algorithm

Input to the procedure: (Data sample) (D))

Output: $H(x)$ = sin($\sum T$ $\alpha ihi(x)$) is the final Hypothesis:

Function: Initialization of the weight $D(i)$ = 1 for i=1 to m.

Begin

For each class i=1 to m over distribution $D(i)$ do Train the weak classifier $D(i)$

Calculate Hypothesis h(i). Weak hypothesis (i).

Return final hypothesis

End

4. RESULTS AND DISCUSSION

The approach was implemented using the Python programming language platform. The proposed NEST approach was analyzed by applying measurements and compares to certain other current models that use DL and hybrid rule-based models. The numbers of correct and wrong outputs were totalled and compared with results of the

reference in a categorization exercise. precision, Accuracy, specificity, recall, and F1-score are among the most commonly used matrices. The NSL-KDD collection contains 77,054 normal records and 71,460 assault documents. The proposed NEST model's effectiveness is contrasted with existing XGBOOST-DNN [15], HNGEA [17], and HDLNIDS [19] methods

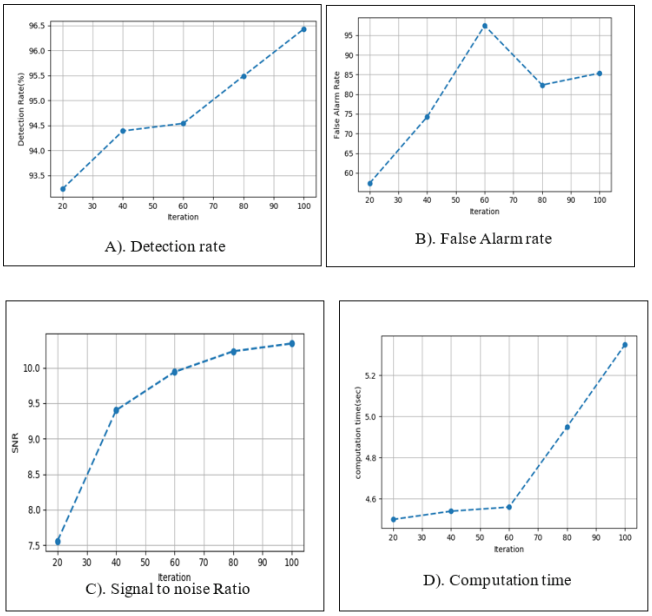


Figure 2. Performance of the BPSO and CFA methods with the GST, Q-learning, Meta- AdaboostM1 classifier for NSL-KDD dataset: A). Detection rate, B). False alarm rate C). Signal to noise Ratio D) Computation time.

Figure 2 displays the Performance of the BPSO and CFA methods with the GST, Q-learning, Meta- AdaboostM1 classifier for NSL-KDD dataset. All detection rates, false alarm rates, signal to noise ratio, specificity, F-measure, and computation time criteria decrease as the number of features grows. Furthermore, when compared to the results obtained using the BPSO and CFA methods, the proposed NEST strategy outperforms them in all classifiers. Furthermore, the use of 20 features produces the best results when compared to alternative numbers of features.

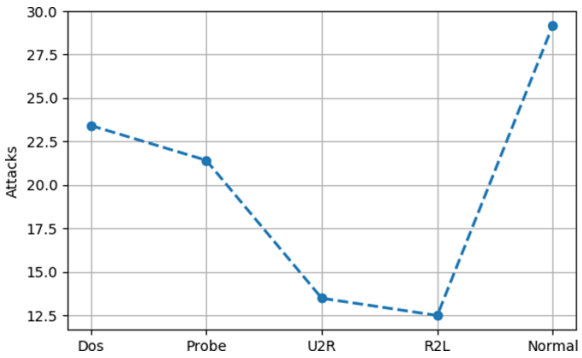


Figure 3. NSL-KDD dataset with Five classes

Figure 3. shows the attack records. To determine the types of records such as DoS, normal, U2R, R2L, and probe, both datasets showed overall higher performance for intrusion detection, even though some results, such as U2R, are not very high.

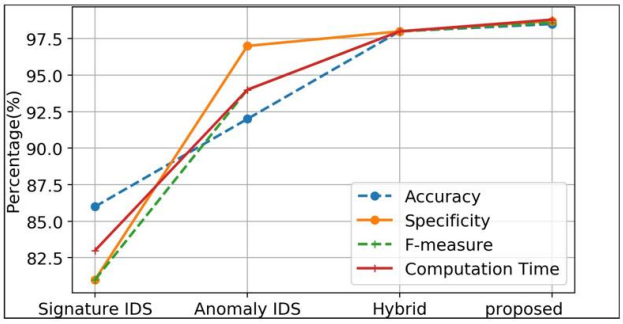


Figure 4. Performance comparison

Figure 4 displays the performance evaluation of our proposed NEST method's three elements.

Table 1. Classes of training and testing: normal and attack

Attack/Normal class	NSL-KDD Train+	NSL-KDD Test+
Normal	67354	7653
DoS	45654	8766
Prob	12237	2537
R2L	679	3678
U2R	34	200
Total	125958	22834

These three subsystems are hybrid detection, anomaly-based, and signature-based intrusion detection systems. The findings of three module employed in the proposed NEST method are shown in Fig. 4. The results also showed that our suggested NEST method outperformed the other three modules in terms of performance (SDM, ADM and HDM). Table 1 shows the Normal and attack class of training and testing set

It's a labelled flow-based dataset that was used to evaluate anomaly-based intrusion detection. There are both normal and attack classes in it. A traffic record is classed as normal, aggressor, unknown, suspicious, or victim in every occurrence.

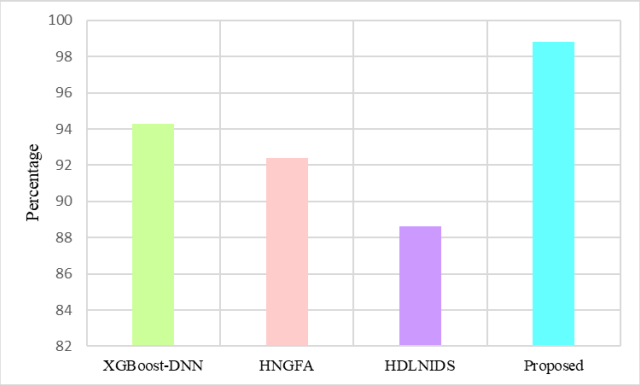


Figure 5. Comparison in terms of detection rate

Figure 5 illustrates a performance comparison of the detection rates between the proposed NEST technique and the existing methods: XGBOOST-DNN [15], HNGEA [17], and HDLNIDS [19]. It assesses how effectively each system or procedure can identify or locate the intended target among all possible targets. The proposed NEST technique demonstrates a higher detection rate compared to the current methods, with improvements of 4.55%, 6.47%, and 10.32% over XGBOOST-DNN, HNGEA, and HDLNIDS, respectively.

5. CONCLUSION

This research discovered a number of intrusion detection issues that put the availability, integrity, and confidentiality of mobile edge networks at jeopardy. To address current intrusion detection difficulties, this work developed a NEST approach for a mobile edge computing environment. Several detection modules using various classifiers make up the suggested detection system. The traffic packets next enter the hybrid IDS phase, which uses signature matching and the GST algorithm to implement SDM. The ADM processes all strange packets, and the deep Q-learning algorithm uses SNR to identify assaults. The detection rate of the proposed NEST method is 4.55%, 6.47%, and 10.32% higher than the existing XGBOOST-DNN, HNGEA, and HDLNIDS, techniques respectively. The ADM is found to outperform earlier IDS approaches after data analysis. This ADM system should be expanded in the future to incorporate the following: Include other major attacks in other datasets and use deep learning methodologies with optimization to test the network's performance. Our technology might be connected to the Instruction Prevention System (IPS), which would automatically protect against deep learning-based assaults. Assure that redirected IoT traffic originates from a registered or unregistered user, and authenticate separate safety using biometric and other authentication mechanisms.

CONFLICTS OF INTEREST

Not applicable.

FUNDING STATEMENT

Not applicable.

ACKNOWLEDGEMENTS

The author would like to express his heartfelt gratitude to the supervisor for his guidance and unwavering support during this research for his guidance and support.

REFERENCES

[1] Z. Azam, M.M. Islam, and M.N. Huda, "Comparative analysis of intrusion detection systems and machine learning based model analysis through decision tree", *IEEE Access.* 2023. [CrossRef] [Google Scholar] [Publisher Link]

[2] Ö. Aslan, S.S. Aktuğ, M. Ozkan- Okay, A.A. Yilmaz, and E. Akin, "A comprehensive review of cyber security vulnerabilities, threats, attacks, and solutions", *Electronics,* vol. 12, no. 6, pp.1333. 2023. [CrossRef] [Google Scholar] [Publisher Link]

[3] S. Vesic, and M. Bjelajac, "Cyber security of a critical infrastructure", *Law Theory & Prac.*, vol. 40, p.77. 2023. [CrossRef] [Google Scholar] [Publisher Link]

[4] D.P. Möller, "Intrusion detection and prevention," In *Guide to Cybersecurity in Digital Transformation: Trends, Methods, Technologies, Applications and Best Practices*, pp. 131-179, 2023. Cham: Springer Nature Switzerland. [CrossRef] [Google Scholar] [Publisher Link]

[5] R. Kumar, K.S. Sangwan, C. Herrmann, and S. Thakur, "A cyber physical production system framework for online monitoring, visualization and control by using cloud, fog, and edge computing technologies", *International Journal of Computer Integrated Manufacturing*, vol. 36, no. 10, pp.1507-1525, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[6] M. Zhu, M. Liang, H. Li, Y. Lu, and M. Pang, "Intelligent acceptance systems for distribution automation terminals: an overview of edge computing technologies and applications", *Journal of Cloud Computing*, vol. 12, no. 1, pp.149, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[7] M. Talebkhah, A. Sali, V. Khodamoradi, T. Khodadadi, and M. Gordan, "Task offloading for edge-IoV networks in the industry 4.0 era and beyond: A high-level view," *Engineering Science and Technology, an International Journal*, vol. 54, pp.101699, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[8] Z. Lv, and W. Shang, "Impacts of intelligent transportation systems on energy conservation and emission reduction of transport systems: A comprehensive review", *Green Technologies and Sustainability*, 1(1), p.100002, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[9] L. Islam, and M.T. Hassan, "Performance Evaluation of Vehicle-Centered Traffic Management Using Fog Computing-based Wireless Network", In *2023 26th International Conference on Computer and Information Technology (ICCIT)*, pp. 1-6, 2023. IEEE. [CrossRef] [Google Scholar] [Publisher Link]

[10] S.M. Rajagopal, M. Supriya, and R. Buyya, "FedSDM: Federated learning based smart decision-making module for ECG data in IoT integrated Edge-Fog-Cloud computing environments", *Internet of Things*, pp.100784, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[11] M. A. Hossain, M. S. Hossain, and R. Karim, "Comprehensive architectural network design based on intrusion detection system," *International Journal of Communication and Information Technology*, vol. *4*, no. 2, pp. 12-19, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[12] D. Shankar, G.V.S. George, J.N. JNSS, and P.S. Madhuri, "Deep analysis of risks and recent trends towards network intrusion detection system," *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 1, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[13] M. Bhavsar, K. Roy, J. Kelly, and O. Olusola, "Anomaly-based intrusion detection system for IoT application." *Discover Internet of Things*, vol. 3, no. 1, pp.5, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[14] D. Papamartzivanos, F.G. Mármol, and G. Kambourakis, "Introducing deep learning self-adaptive misuse network intrusion detection systems", *IEEE access*, vol. 7, pp.13546-13560, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[15] P. Devan, and N. Khare, "An efficient XGBoost–DNN-based classification model for network intrusion detection system." *Neural Computing and Applications*, vol. 32, no. 16, 12499-12514, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[16] S. Venkatraman, and B. Surendiran. "Adaptive hybrid intrusion detection system for crowd sourced multimedia internet of things systems", *Multimedia Tools and Applications*, vol. 79, no. 5, pp. 3993-4010, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[17] R. Elhefnawy, H. Abounaser, and A. Badr, "A hybrid nested genetic-fuzzy algorithm framework for intrusion detection and attacks," *IEEE Access*, vol. 8, pp.98218-98233, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[18] W. Seo, and W. Pak, "Real-time network intrusion prevention system based on hybrid machine learning", *IEEE Access*, vol. 9, pp.46386-46397, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[19] E.U.H. Qazi, M.H. Faheem, and T. Zia, "HDLNIDS: hybrid deep-learning-based network intrusion detection system", *Applied Sciences*, vol. 13, no. 8, pp.4921, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[20] V. Hnamte, H. Nhung-Nguyen, J. Hussain, and Y. Hwa-Kim, "A novel two-stage deep learning model for network intrusion detection: LSTM-AE", *IEEE Access*. 2023. [CrossRef] [Google Scholar] [Publisher Link]

AUTHORS

A. Biju is an Assistant Professor in Maria College of Engineering and Technology, Attoor. He is the Head of the Master of Computer Applications Department. He received his BE and ME degrees in Anna University, Chennai, in 2006 and 2011, respectively.

S. Wilfred Franklin Professor and Head in the Department of Electronics and Communication Engineering, CSI Institute of Technology, Thovalai, Kanyakumari District, Tamil Nadu, India.

Arrived: 18.09.2024
Accepted: 21.10.2024

International Journal of System Design and Computing (IJSDC)
Volume 02, Issue 02, July - December (2024)

RESEARCH ARTICLE

PREDICTIVE MONITORING FRAMEWORK FOR ENHANCING OPERATIONAL RESILIENCE IN RETAIL FULFILLMENT SYSTEMS: A CASE STUDY

Ramya Thatikonda [1, *]

[1]Software Engineer, PhD in Information Technology, University of the Cumberlands, Williamsburg, USA.

[*]*Corresponding e-mail: ramya.t0211@gmail.com*

Abstract – This paper presents the development and transformative impact of an innovative unified predictive monitoring framework designed for a large-scale retail fulfillment system. By seamlessly integrating cutting-edge industry-standard tools such as Grafana, Splunk, Kibana, and Elasticsearch, the framework provides unprecedented real-time visibility into fulfillment operations. The system leverages advanced machine learning algorithms, including time series forecasting, anomaly detection, and classification models, to proactively identify and resolve potential issues, particularly during high-demand periods. This data-driven approach has dramatically improved system stability, reducing service interruptions by 30% and enhancing customer satisfaction scores by 15%. The framework sets new benchmarks for operational resilience in the retail sector, demonstrating the transformative power of predictive analytics in managing complex fulfillment systems at scale.

Keywords *– Predictive Monitoring; Operational Resilience; Retail Fulfillment Systems; Machine Learning.*

1. INTRODUCTION

Retail fulfillment systems are the backbone of modern e-commerce, processing orders, managing inventory, and orchestrating deliveries. As the scale of e-commerce continues to grow, ensuring operational stability during high-traffic periods, such as sales events or holidays, becomes increasingly critical. Retailers are adopting advanced monitoring systems to ensure that fulfillment operations remain efficient and resilient.

This paper explores the architecture of the framework, which integrates Grafana for data visualization, Splunk for real-time analytics, and Kibana with Elasticsearch for log management and search capabilities. The framework is enhanced by machine learning models that predict potential system failures and bottlenecks. We will discuss the implementation of this system, its impact on operational performance during critical retail events, and the broader implications for the retail industry's approach to system monitoring and management.

2. BACKGROUND AND RELATED WORKS

Because of its critical role in ensuring the seamless flow of products, services, and information in a globalised corporate environment, supply chain risk management has garnered a lot of interest from scholars and practitioners in the area [1]. Here we take a look at a comprehensive review of the literature that has looked at different methods, tools, and technologies that have been developed to make supply chains more resilient and agile. Studying the existing body of knowledge in this field allows us to get vital insights into the evolution of risk management systems, discover new trends, and detect the gaps that spurred the construction of the recommended framework. By analysing various perspectives and empirical data, this review provides the framework for understanding the broader context to which our research contributes novel insights and advancements. An abundance of literature has shed light on the many facets, potential benefits, and current challenges of integrating state-of-the-art technology into supply chain risk management [2- 4]. In this part, we carefully examine the academic efforts that have helped us understand this ever-changing landscape, paying special attention to their methods, findings, and the gaps that they have found.

The study conducted by Wong et al. [5] mostly focused on SMEs. A structural model was developed by them that integrated supply chain re-engineering, agility, and AI-driven risk management abilities. Using partial-least-squares-based structural equation modelling (PLS-SEM) and artificial neural network (ANN) techniques, the study found that AI has a positive effect on risk management, re-engineering abilities, and the intermediary function these skills play in improving supply chain agility. This research provided valuable insights on the use of AI in addressing demand uncertainty, allowing for more informed decision-making and efficient allocation of resources.

Giannakis and Louis presented a novel approach to supply chain management that relies on multi-agent systems and is backed by big data analytics [6]. The goal of

developing this framework was to influence supply chain agility by means of autonomous control actions for corrective purposes. The study's three primary findings—responsiveness, flexibility, and speed—formed the basis for the system's organisational design. The results of this study show that state-of-the-art technology may increase agility in all areas, which means that supply chain operations can be more responsive, flexible, and fast to respond.

Jayender and Kundu [7] provide insights on the factors impacting the agility of the automotive industry's supply chain. By looking at the potential for interoperability between big data analytics and ERP systems, their study aimed to determine how this relationship impacts the industry's ability to stay agile. To address implementation issues and increase agility, a graph-theory-based approach was proposed. In order to ensure agility in complex enterprises, this research demonstrated the need for new tactics.

In his study, Shamout [8] looked at the relationship between supply chain data analytics and agility. The study used fuzzy sets qualitative comparative analysis (fsQCA) to assess causal recipes that predict high levels of supply chain agility based on a combination of supply chain data analytics, company size, firm age, and yearly sales. This research emphasised the significance of comprehensive data analytics and how it helps achieve supply chain agility in a complex corporate environment.

In order to assess the use of machine learning (ML) for SCRM, Schroeder and Lodemann [9] conducted a comprehensive literature review. They examined the theoretical and practical applications of ML in handling supply chain risks, particularly in recognising manufacturing, transport, and supply hazards. By integrating new data sources and offering in-the-moment insights into possible threats, the research illustrated how ML may enhance SCRM.

In order to enhance supply chain visibility and proactively decrease risks, Lee et al. [10] examined real-time event detection using Twitter information and blockchain technology. This innovative approach showcased how blockchain technology, together with real-time data collection, might improve risk management.

Ganesh and Kalpana [11] conducted a comprehensive literature review that examined the use of AI and ML approaches across all phases of supply chain risk management. It became clear from the study which AI algorithms were utilised and which supply chain concerns were addressed. This study identified gaps in the current literature, proposed interesting avenues for further research, and highlighted both the opportunities and challenges for implementation.

Ivanov and Dolgui were the first to suggest the concept of a digital supply chain twin [12]. This would be a computer model that mimics the states of the network in real time. The study's focus was on supply chain visibility and interruption risk management via the use of digital twins. The COVID-19 pandemic and its effects on supply chains highlighted the critical need of digital twins for company continuity.

Mageto [13] used Toulmin's reasoning paradigm to investigate the connection between sustainable supply chain management and big data analytics. The paper highlighted the ways in which big data analytics enhances sustainable practices within industrial supply chains and identified concerns such as skill shortages and cyberattacks. Dolgui and Ivanov [14] investigated how 5G may enhance smart operations and digital supply chains. They found possible prospects for change across operational processes and strategic views. The research also addressed the pros and cons of 5G technology adoption.

Retailers require robust monitoring systems to ensure that fulfillment processes run smoothly, especially during peak shopping events. Traditional monitoring approaches, which often rely on static thresholds and manual intervention, are insufficient for detecting and mitigating potential issues in real-time. For example, simple rule-based alerts on metrics like CPU utilization or network latency often fail to capture complex system behaviors that lead to failures.

Previous research in retail systems monitoring has shown the benefits of real-time analytics and predictive modeling Ahilan et al. (2023) demonstrated a 20% improvement in issue detection time using real-time analytics in a mid-sized retail environment [15]. Sivasankari et al. (2024) explored the use of machine learning for demand forecasting in retail supply chains, achieving a 15% reduction in stockouts [16].

However, many existing solutions fall short in scalability and precision during high-demand events [17]. The complexity of modern retail systems, with their interconnected microservices and distributed architectures, requires a more sophisticated approach. This paper builds on these foundational works by introducing a unified system that integrates multiple monitoring tools alongside advanced machine learning algorithms. Our approach addresses the gap in the literature by demonstrating how a comprehensive, ML-driven monitoring framework can be implemented and scaled in a large retail environment, providing proactive issue resolution even during the most demanding retail events.

3. METHODOLOGY: DESIGN OF THE UNIFIED MONITORING FRAMEWORK

3.1. System Architecture

Because The architecture of the predictive monitoring framework is built around the integration of several open-source tools. Each tool serves a specific purpose, contributing to the framework’s ability to monitor system health, perform data analytics, and offer predictive insights:

- **Grafana**: Used for real-time data visualization, displaying key performance indicators (KPIs) and operational metrics.
- **Splunk**: An analytics tool that processes large volumes of log data and uses machine learning to detect anomalies.

- **Kibana and Elasticsearch**: Used for log management, with Kibana providing an interactive interface for data exploration and Elasticsearch handling data indexing and querying.

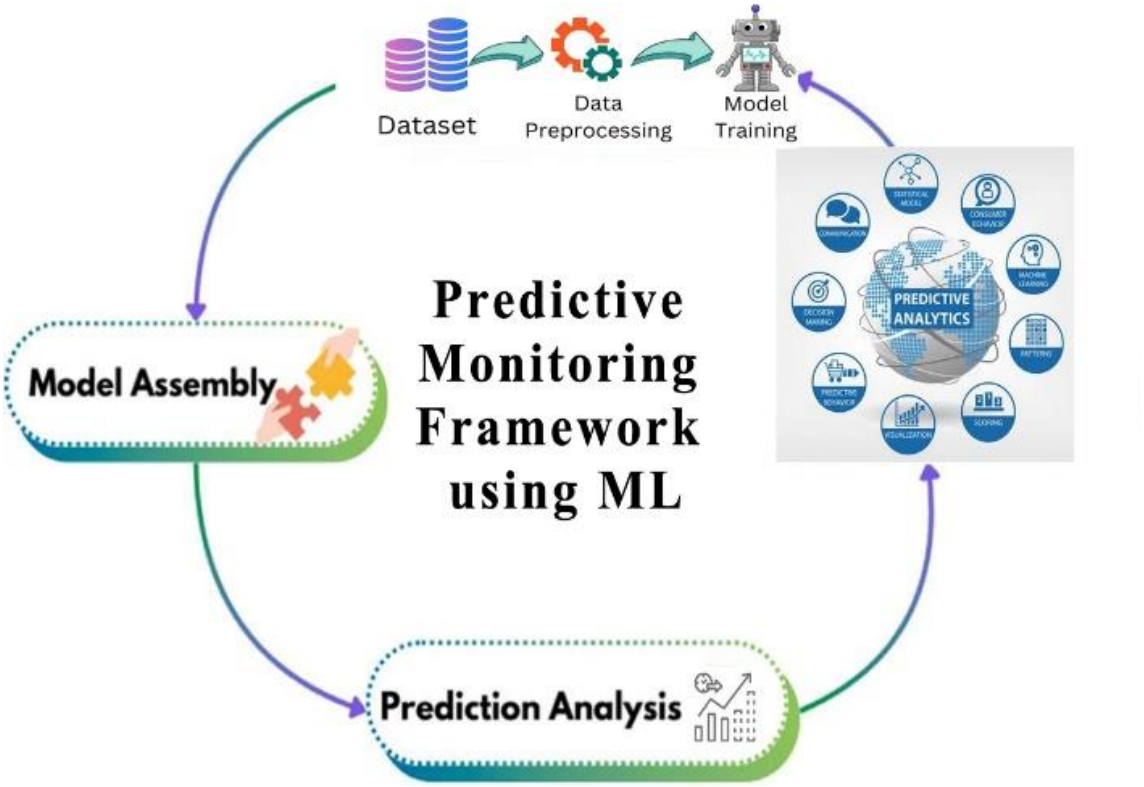


Figure 1. System Architecture Overview

In the figure 1. above, the system's architecture includes data ingestion from fulfillment applications, real-time processing, and output to Grafana dashboards and Splunk for predictive analysis.

3.2. Data Integration and Real-Time Monitoring

The system collects data from various sources, including order management systems, inventory databases, and fulfillment processing systems. This data is ingested into Kafka streams, which act as a central nervous system for the framework. From Kafka, data is simultaneously fed into Elasticsearch for indexing and long-term storage, and into Splunk for real-time analysis.

The real-time dashboards, generated through Grafana, give operational teams insights into system health, enabling rapid response to potential issues. These dashboards are customizable and include features such as heat maps for geographical order distribution and time series graphs for key performance indicators.

Key metrics monitored include:

- **Order Processing Time**: The time it takes for an order to be processed from receipt to dispatch.
- **Inventory Levels**: Monitoring stock availability to prevent out-of-stock situations.
- **Shipment Delays**: Measuring the time between order fulfillment and delivery.

Table 1. Key Metrics and Thresholds

Metric	Description	Threshold
Order Throughput	Orders processed per minute	5000 orders/min
System Latency	Time taken for order processing	100 ms
Network Latency	Time for inter-service communication	50 ms
Resource Utilization	CPU, memory usage of servers	85% CPU, 80% RAM

3.3. Machine Learning for Predictive Analytics

Time Series Forecasting: Using ARIMA (Auto Regressive Integrated Moving Average) models to predict future values of key metrics based on historical patterns. This allows the system to anticipate spikes in order volume or processing times.

Anomaly Detection: Implementing Isolation Forest algorithms to identify unusual patterns in system behavior that may indicate impending issues. This helps in detecting subtle deviations that might be missed by traditional threshold-based monitoring.

Classification Models: Utilizing Random Forest classifiers to categorize system states and predict the likelihood of specific types of failures. This enables the system to provide targeted alerts and recommendations.

These ML algorithms are trained on historical data, including logs from previous peak events, and are

continuously updated with new data to improve their accuracy over time. The models focus on identifying patterns in:

- Order volume fluctuations
- Processing time anomalies
- Resource utilization trends
- Error rate variations

By proactively flagging potential issues before they escalate, the framework significantly reduces the likelihood of service interruptions. The ML models generate alerts that are seamlessly integrated into the existing monitoring dashboards, providing actionable insights to the operations team.

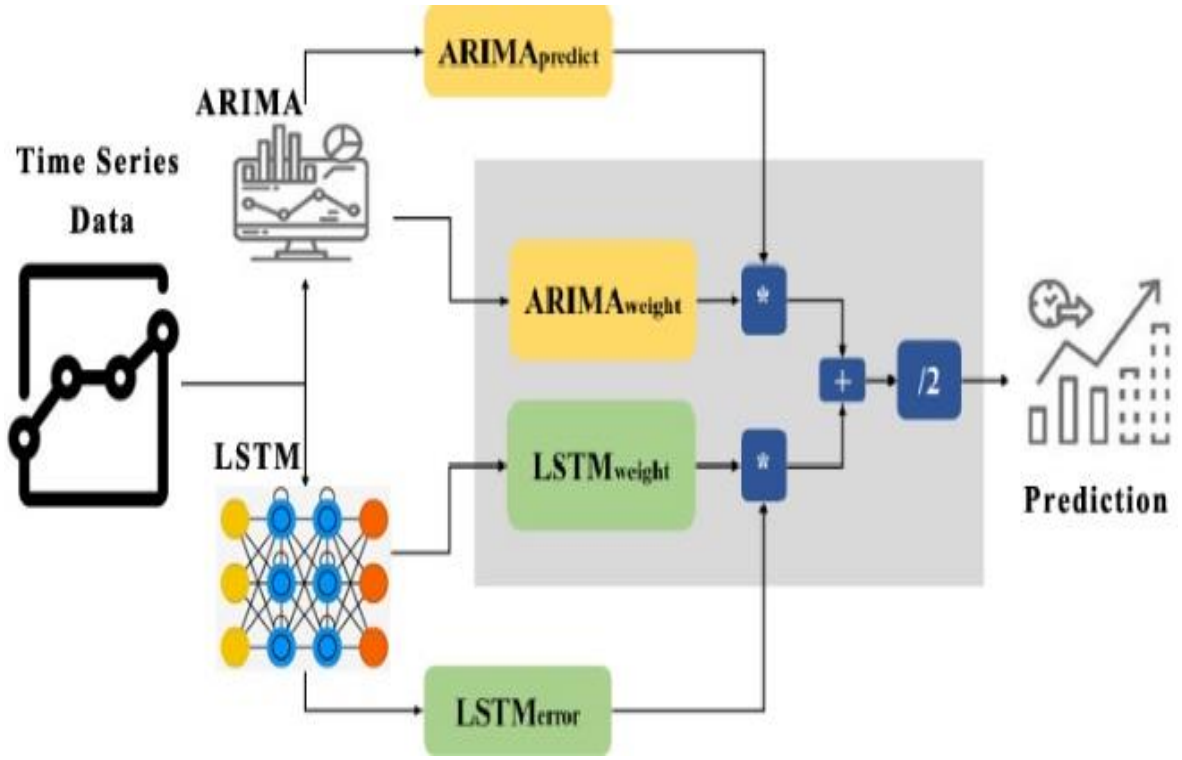


Figure 2. Machine Learning Model Workflow

The figure 2. above illustrates how the system uses time-series forecasting models (ARIMA, LSTM) for predictive analytics. These models are integrated into Splunk to provide proactive alerts about potential system issues.

4. PREDICTIVE MONITORING AND MULTI-TEAM INTEGRATION CHALLENGES

In a large-scale retail fulfillment system, the order fulfillment process involves multiple teams, systems, and stages. Each team or system performs a specific function that contributes to the overall order processing workflow. However, given the interconnectedness of these processes, a single failure in one part of the system can cascade and cause disruptions, delays, or errors. Below is a breakdown of the stages involved in the order processing lifecycle and potential issues that may arise at each stage.

4.1. Order Placement and Fraud Checks

- **Team(s) Involved**: Customer service, fraud prevention teams, IT security.
- **Key Tasks**: Order creation, customer payment verification, fraud checks, and approval.

Potential Issues:

- **Payment Failures**: A customer's payment might fail due to incorrect billing information, insufficient funds, or payment gateway errors.
- **Fraudulent Orders**: Fraud detection systems might flag legitimate orders incorrectly, leading to delays or cancellations.

Predictive Monitoring Role:

- The predictive system can track and alert teams to a sudden spike in transaction failures, potentially indicating system overloads or malicious activity.
- Machine learning models can help predict high-risk orders by analyzing historical data, flagging potential fraud attempts in real-time.

Example Alert:

- **High Fraud Rate Detected**: "There has been a 40% increase in flagged orders this hour compared to historical trends. Investigate."

4.2. Sourcing and Order Management Systems (OMS)

- **Team(s) Involved**: Sourcing, inventory management, OMS.

- **Key Tasks**: Allocating inventory based on customer order, checking stock levels, confirming sourcing channels (warehouse, vendors), and updating the OMS with real-time data.

Potential Issues:

- **Out-of-Stock Items**: Items may be out of stock or incorrectly updated in the OMS, leading to order delays or cancellations.
- **Incorrect Inventory Levels**: Discrepancies between the available inventory and what is recorded in the system can result in failed orders.
- **Sourcing Delays**: Delay in sourcing due to vendor issues or delays at specific warehouses.

Predictive Monitoring Role:

- **Inventory Monitoring**: The system can flag products with low stock levels and alert sourcing teams to take corrective action, reducing the risk of out-of-stock issues.
- **Lead Time Predictions**: Predictive analytics models can estimate delays based on past sourcing and warehouse times, providing visibility to the sourcing team.

Example Alert:

- **Stock Level Alert**: "Product XYZ is low on stock in Warehouse A. Predicted out-of-stock time within 2 hours. Please restock immediately."

4.3. Promise Delivery Times and Fulfillment Management

- **Team(s) Involved**: Delivery promise teams, fulfillment management team, operations teams.
- **Key Tasks**: Promising delivery dates to customers, optimizing fulfillment routes, scheduling deliveries.

Potential Issues:

- **Missed Delivery Promises**: Delays in order processing or logistics can cause promised delivery times to be missed, leading to poor customer satisfaction.
- **Route Optimizations Failures**: Issues in the fulfillment system may prevent the optimization of delivery routes, causing delivery delays or increased costs.

Predictive Monitoring Role:

- Predictive analytics can monitor delivery times and alert fulfillment managers if the system detects that an order might miss its promised delivery time.
- **Real-time Data Visualization**: Using Grafana or Kibana, the team can track delivery progress and see deviations from the schedule in real time.

Example Alert:

- **Delivery Time Prediction**: "Order 12345 scheduled for delivery in 3 hours is at risk of delay due to traffic congestion in Region X. Alternative routes suggested."

4.4. Delivery Teams and Return

***s*Team(s) Involved**: Last-mile delivery teams, returns management teams, customer support.

- **Key Tasks**: Delivering the order to the customer, handling any return requests.

Potential Issues:

- **Last-Mile Delays**: Issues such as traffic, weather conditions, or incorrect addresses can delay the last-mile delivery process.
- **Returns and Customer Support**: If a customer decides to return the product, returns processing can add complexity to the order lifecycle, affecting inventory accuracy and customer satisfaction.

Predictive Monitoring Role:

- **Last-Mile Monitoring**: The predictive system can integrate real-time traffic data to warn delivery teams about potential delays and suggest alternative routes.
- **Returns Prediction**: By analyzing historical data, the system can predict the likelihood of returns based on factors such as product type, delivery region, and customer behavior.

Example Alert:

- **Delivery Delay Prediction**: "Customer delivery for Order ID 56789 is delayed by 20 minutes due to adverse weather conditions. Estimated new delivery time: 1:45 PM."

4.5. Cross-Team Communication and Dependencies

- **Teams Involved**: Multiple cross-functional teams including Order Management, Fraud, Sourcing, Delivery, and Returns.

Potential Issues:

- **Communication Breakdowns**: Delays or errors can happen if teams do not effectively communicate or share real-time updates about order status.
- **System Integration Failures**: If different systems (OMS, fraud check, fulfillment systems) are not properly integrated, they can cause delays or mismatches in data.

Predictive Monitoring Role:

- **Cross-Team Visibility**: By creating a centralized dashboard using Grafana, all teams can have real-time visibility into the status of every order, reducing silos and ensuring effective communication.

- **Automated Notifications**: The system sends automated alerts to teams if there's a communication breakdown or delayed information across systems.

Example Alert:

- **System Integration Alert**: "Order management system not updated with new inventory levels from the fulfillment center. Please investigate the system sync delay."

5. PREDICTIVE ANALYTICS IN ACTION: REAL-WORLD EXAMPLES FROM RETAIL ORDER PROGRAMS

5.1. Example Dataset: Express Order Program

In The Express Order Program processes orders that need to be delivered within 24 hours. Predictive analytics was applied to monitor the program's performance during a high-demand event. The Example Dataset for Express Orders is provided in Table 2.

Table 2. Example Dataset for Express Orders

Order ID	Order Received Time	Order Fulfilled Time	Shipment Status	Resource Utilization (%)	Network Latency (ms)	Predicted Delay (minutes)
12345	2023-11-15 10:00 AM	2023-11-15 11:30 AM	Shipped	75%	45	0
12346	2023-11-15 10:05 AM	2023-11-15 11:45 AM	Shipped	80%	50	5
12347	2023-11-15 10:10 AM	2023-11-15 12:00 PM	Pending	85%	60	10

The predictive model correctly flagged Order ID 12347 as likely to face a delay due to high resource utilization and network latency, allowing proactive measures to be taken.

5.2. Example Dataset: Scheduled Order Pickup Program

The Scheduled Order Pickup Program involves customers picking up their orders at physical stores. The program relies on accurate inventory and timely order processing. An Example Dataset for Scheduled Pickup Orders is given in Table 3.

Table 3. Example Dataset for Scheduled Pickup Orders:

Order ID	Pickup Time	Order Fulfilled Time	Order Status	Inventory Level	Predicted Delay (minutes)
23456	2023-11-15 1:00 PM	2023-11-15 12:30 PM	Ready for Pickup	12	0
23457	2023-11-15 2:00 PM	2023-11-15 1:50 PM	Ready for Pickup	8	5
23458	2023-11-15 3:00 PM	2023-11-15 3:20 PM	Pending	0	15

The system flagged Order ID 23458, which faced a shortage, allowing inventory to be replenished in advance, avoiding customer dissatisfaction.

6. PREDICTIVE MONITORING AND MULTI-TEAM INTEGRATION CHALLENGES

By integrating predictive monitoring and machine learning models across all stages of the order fulfillment lifecycle, the system can:

- **Anticipate Issues**: Predict when and where problems are likely to occur (e.g., fraud spikes, out-of-stock items, delivery delays).
- **Optimize Response Times**: Reduce incident response times and ensure faster corrective action.

Enhance Communication: Improve collaboration between cross-functional teams by providing real-time data, alerts, and automated workflows.

7. CONCLUSION

The predictive monitoring framework has proven instrumental in elevating operational resilience, particularly during peak demand periods in retail. By integrating tools such as Grafana, Splunk, Kibana, Elasticsearch, and machine learning algorithms, this system has effectively anticipated and mitigated potential disruptions, maintaining system stability, reducing downtime, and ultimately enhancing customer satisfaction. The synergy of machine learning and advanced data analytics in predictive monitoring sets a new benchmark for managing the complexities of large-scale e-commerce order fulfillment systems, offering proactive insights that drive operational agility and reliability.

While this framework has already achieved substantial improvements in system resilience, future developments are set to push its capabilities even further. Real-time data augmentation is a key focus, aiming to incorporate external factors such as weather patterns or regional disruptions to enhance predictive accuracy and provide richer context for operational decisions. Scalability is also a priority, with plans to expand the system's reach to monitor global fulfillment centers in real time, thereby offering a cohesive, interconnected approach to operational oversight. Additionally, the incorporation of deep reinforcement learning holds exciting potential, with experiments underway to explore how reinforcement learning techniques could facilitate dynamic decision-making in resource allocation. This advanced approach would enable the system to learn and adapt to real-time changes, further optimizing performance in high-stakes, high-demand scenarios. Together, these advancements position predictive monitoring

as a powerful tool for future-ready retail operations, where agility and precision are essential to meet customer expectations and operational demands.

CONFLICTS OF INTEREST

The authors declare that there is no conflict of interest.

FUNDING STATEMENT

Authors did not receive any funding.

ACKNOWLEDGEMENTS

The author would like to express his heartfelt gratitude to the supervisor for his guidance and unwavering support during this research for his guidance and support.

REFERENCES

[1] A. Gunasekaran, K.H. Lai, and T.E. Cheng, "Responsive supply chain: A competitive strategy in a networked economy", *Omega*, vol. 36, pp. 549–564, 2008 [CrossRef] [Google Scholar] [Publisher Link]

[2] R. Rawat, "A Systematic Review of Blockchain Technology Use in E-Supply Chain in Internet of Medical Things (Iomt)", *Int. J. Comput. Inf. Manuf.*, vol. 2, no. 2, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[3] N.A. Perifanis, and F. Kitsios, "Investigating the influence of artificial intelligence on business value in the digital era of strategy: A literature review", Information, vol. 14, no. 85, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[4] S. Srivastava, "Optimizing Supply Chain with AI and Analytics", Appinventiv, 2023. [Online]. Available: https://appinventiv.com/blog/ai-in-supply-chain-analytics/[CrossRef] [Google Scholar] [Publisher Link]

[5] L.W. Wong, G.W.H. Tan, K.B. Ooi, B. Lin, and Y.K. Dwivedi, "Artificial intelligence-driven risk management for enhancing supply chain agility: A deep-learning-based dual-stage PLS-SEM-ANN analysis", Int. J. Prod. Res., pp. 1–21, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[6] M. Giannakis, and M. Louis, "A multi-agent based system with big data processing for enhanced supply chain agility", J. Enterp. Inf. Manag., vol. 29, pp. 706–727, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[7] P. Jayender, and G.K. Kundu, "Intelligent ERP for SCM agility and graph theory technique for adaptation in the automotive industry in India", Int. J. Syst. Assur. Eng. Manag., pp. 1–22, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[8] M.D. Shamout, "Supply chain data analytics and supply chain agility: A fuzzy sets (fsQCA) approach", Int. J. Organ. Anal., vol. 28, pp. 1055–1067, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[9] M. Schroeder, and S. Lodemann, "A systematic investigation of the integration of machine learning into supply chain risk management", Logistics, vol. 5, no. 62, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[10] C.H. Lee, H.C. Yang, Y.C. Wei, and W.K. Hsu, "Enabling blockchain-based SCM systems with a real-time event monitoring function for preemptive risk management", Appl. Sci., vol. 11, no. 4811, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[11] A.D. Ganesh, and P. Kalpana, "Future of artificial intelligence and its influence on supply chain risk management–A systematic review", Comput. Ind. Eng., vol. 169, p. 108206, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[12] D. Ivanov, and A. Dolgui, "A digital supply chain twin for managing the disruption risks and resilience in the era of Industry 4.0", Prod. Plan. Control, vol. 32, pp. 775–788, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[13] J. Mageto, "Big data analytics in sustainable supply chain management: A focus on manufacturing supply chains," Sustainability, vol. 13, no. 7101, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[14] A. Dolgui, and D. Ivanov, "5G in digital supply chain and operations management: Fostering flexibility, end-to-end connectivity and real-time visibility through internet-of-everything", Int. J. Prod. Res., vol. 60, pp. 442–451, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[15] A. Ahilan, J. Angel Sajani, A. Jasmine Gnana Malar, and B. Muthu Kumar, "Machine Learning-Based Brain Disease Classification Using EEG and MEG Signals", In *International Conference on Frontiers of Intelligent Computing: Theory and Applications*, pp. 487-498, 2023, April. Singapore: Springer Nature Singapore. [CrossRef] [Google Scholar] [Publisher Link]

[16] B. Sivasankari, A. Ahilan, S. Rajakumar, and R. Raja Kumar, "Predicting ozone depletion using BIn2O3 with RH based solid conductometric sensor", 2024. [CrossRef] [Google Scholar] [Publisher Link]

[17] A. Appathurai, and P. Deepa, "Radiation induced multiple bit upset prediction and correction in memories using cost efficient CMC," *Informacije MIDEM*, vol. 46, no. 4, pp. 257-266, 2016. [CrossRef] [Google Scholar] [Publisher Link]

AUTHORS

Ramya Thatikonda is a distinguished software engineer at Walmart, leveraging her expertise in Information Technology to drive innovation and efficiency in one of the world's largest retail corporations. With a Ph.D. in Information Technology and a Master's degree in Computer Information Systems, she stands at the forefront of technological advancement. Her academic journey has been marked by a profound dedication to research, focusing on cutting-edge fields such as Blockchain, Artificial Intelligence, and Machine Learning. Dr. Thatikonda's contributions extend beyond the academic realm, with her research findings published in renowned international journals, enriching the global discourse on emerging technologies. With over a decade of experience in the Information Technology industry, Dr. Thatikonda has honed her skills in requirement analysis, design, and development of database solutions across diverse sectors including Healthcare, Retail, Supply Chain, and E-Commerce. Her expertise has been instrumental in spearheading projects ranging from migration/conversion initiatives to application development, data analysis, and process automation, driving tangible outcomes and elevating organizational performance. As a seasoned IT professional, Dr. Thatikonda exemplifies a commitment to excellence and innovation, continuously pushing the boundaries of technological possibilities to create impactful solutions that resonate across industries.

Arrived: 25.09.2024

Accepted: 01.10.2024

International Journal of System Design and Computing (IJSDC)
Volume 02, Issue 02, July - December (2024)

RESEARCH ARTICLE

HYBRID NEURAL SYSTEM FOR CYBER-ATTACK DETECTION IN LARGE-SCALE SMART GRIDS

R.A. Mabel Rose [1, *] and V. Padmajothi [2]

[1] Department of Computer Science and Business Systems, Panimalar Engineering College, An Autonomous Institution, Affiliated to Anna University, Chennai, Bangalore Trank Road, India.

[2] Department of Electronics and Communication Engineering, SRM Institute of Science and Technology (SRMIST), Kattankulathur, Tamil Nadu 603203, India.

[*]*Corresponding e-mail: mabel.nidhu@gmail.com*

Abstract – The advancements in the distributed energy system and digital technology in the smart grid system increase efficiency, stability, and reliability. However, it increases the vulnerabilities in the grid network. The falsely injected data in the grid network leads to failures in energy production, and consumption. Hence, an Ant Lion-based Modular Neural network (ALbMNN) model was proposed to detect the normal and malicious data. The presented model integrates the ant lion fitness and MNN attribute to detect the falsely injected data in the grid system. The dataset was initialized and pre-processed using the divide-and-conquer principle of MNN. The optimal ant lion fitness solution helps in selecting features optimally from the dataset. Finally, the presented model was assessed with a large-scale smart grid dataset, and the results are estimated. Moreover, a comparative analysis was performed to verify the performance of the developed scheme. Based on performance and comparative analysis, the suggeated model performed better than other existing methods.

Keywords *– False Data Injection, Cyber-attack detection, Smart grids, Modular Neural Network, Ant Lion Optimization.*

1. INTRODUCTION

The power system consists of a grid of generators and sensors that permits two-way communication through the incorporation of Distributed Energy Resources [1]. This advanced power system has numerous advantages including increased energy reliability, efficiency, and network stability [2]. However, it increases the possibility of cyber-attack on the network due to a high number of associated network devices [3]. The cyber-attack in the smart grid leads to damage to components and causes false demand requests. The false demand request leads to increased energy production, and wastage [4]. In addition, it enables the power system to provide stable energy flow, and reliable performance [5]. Therefore, a continuous monitoring system must be designed to detect/ identify malicious events in the smart grid system [6]. Generally, in a large-scale smart grid system, individual sensors are the major target of security concessions [7]. To avoid false demand requests and cyber-attack in the grid system, advanced technology is developed. Initially, cryptography-based authentication techniques are developed to track and eliminate cyber-attacks [8]. But it does not apply to large-scale smart grids because of its high resource usage, and computational time [9]. The advanced malicious event detection approaches employ state estimation methodology to estimate cyber-attacks efficiently [10]. However, these techniques are mathematically expensive, and not scalable to large-scale grid networks. Moreover, the advancement in the grid network demands a real-time attack supervising system [11].

The conventional intrusion detection scheme cannot access the data from large-scale grid networks [12]. Exploring this huge collected data requires machine learning (ML) and deep learning (DL) methods to process the complex data structure for identifying the false demand request [13]. The cyber-attack detection mechanism identifies the malicious event by learning the complex data structure pattern through non-linear ML [14]. In recent times, studies based on ML-based cyber security are increasing day by day [15]. This scheme deploys a feature selection mechanism to test the dataset attributes. Different machine models like random forest algorithm, decision tree algorithm [16], gradient boost algorithm [17], etc., are applied to handle the complex data. This method maintains scalability and robustness of attack detection by processing high-dimensional data in this way [18].

However, the learning process requires high resource usage and increases the computational burden [19]. The performance comparison of ML and DL-based intrusion detection mechanism shows that the DL-based techniques reduce the computational burden and increases the accuracy in cyber-attack detection [20]. Thus, various DL algorithms like distributed data-driven intrusion detection schemes [21], real-time-based false data injection (FDI) attack detection [22], extremely randomized tree approach [23], deep

learning-based detection methods [24], etc., are developed to detect a malicious event in the smart grid system. However,

they face issues with implementation costs and data processing. Hence, an optimized DL-based cyber-attack detection approach was developed in this article.

The main contribution of the proposed research is described as follows,

- Smart grid datasets were collected from standard sites and imported into the system.
- Develop the optimized neural system (ALbMNN) to detect the falsely injected data.
- Initially, the dataset is pre-processed and then the features are selected to detect malicious events.
- The anti-Lion fitness solution in the proposed model effectively selects the features and increases the system's performance.
- Furthermore, the performance of the proposed model is determined and evaluated in terms of accuracy, true positive rate, and false positive rate.

The presented article is sequenced as follows, the works related to the cyber-detection in smart grid is described in 2nd section, the problems in the existing techniques are illustrated in 3rd section, the proposed model is explained with a flowchart and algorithm in 4th section, the results of the developed model is detailed in 5th section, and the conclusion of the article is mentioned in 6th section.

2. RELATED WORKS

Following are some recent articles about cyber-attack detection in smart grid systems:

Jiayu Shi et al. [21] suggested a distributed data-driven intrusion detection scheme to identify the malicious event/FDI in the smart grids. Moreover, it helps in preventing the over-fitting issue which usually occurs in ML algorithms. Further, the developed model is simulated and verified with other conventional models. Although this model reduces the overfitting issue and estimates the malicious event precisely, it is highly expensive and requires more resources.

However, the FDI attack has become a serious threat to the state estimation approach Injecting malicious data into real-time data affects system performance. Thus, Debottam Mukherjee et al. [22] presented a real-time-based FDI attack detection model to ensure security in the grid system. The performance of this algorithm shows that it processes real-time data with minimum error. This provides highly accurate intrusion detection with less error using the error covariance matrix. However, the computational burden is high in this model.

However, the incorporation of digital communication techniques leads to vulnerabilities in the smart grid system. Hence, Seyed HosseinMajidi et al. [23] designed an approach to detect the FDI attack optimally. Further, the results obtained by this model are compared with traditional algorithms like support vector machine (SVM), Decision tree, random forest, k-nearest neighbor (KNN), etc., to verify the system performance. However, the system size and computational complexity are high in this model.

At the same time, the vulnerabilities are increasing because of the increased number of devices. Yucheng Ding et al. [24] reported a DL-based detection method to investigate information corruption. This conditional-based DL scheme analyzes and trains the huge input dataset to predict malicious events accurately. The developed scheme is validated using the IEEE standard test system. But the designed model reduces the reliability performance of the grid system.

The fully developed distributed and automated electricity grid system growths the possibility of cyber-attacks. The cyber-attacks increase the false demand request leading to the wastage of energy in smart grids. Moreover, the growth of cyber threats reduces grid reliability performances by injecting false data into it. Therefore, SudhakarSengan et al. [25] present the combination of true data integrity in the physical layers. This feedback-based network improves the FDI attack detection rate. Although the developed model earned 98.19% accuracy, the data processing consumes more time and size.

The data acquisition and supervisory control mechanism in the grid system enables the hacker to inject bas-data. This leads to huge energy and financial losses in the grid network. To overcome these cyber threats, Mario R. Camana Acosta et al. [26] suggested an effective intrusion detection model based on kernel and randomized tree principles. This technique reduces the component size, and system complexity. Additionally, for verification purposes, the outcomes are contrasted with the most advanced methods. However, it is not resistant to other types of attacks related to smart grids.

The power sector is rising as the major energy source across the world. The increasing energy demand and wide usage of network-connected devices pave the way for security threats. The security threats in the grid system cause corruption in power transmission, false demand requests, and energy wastage. To prevent these challenges, Yangyang Tian et al. [27] presented a hybrid model based on a different machine and DL schemes. This method uses a CNN and SVM techniques for classifying malicious attacks. This intrusion detection algorithm effectively reduces the over-fitting issue and identifies the damage in transmission and production lines in smart grids. The effectiveness of this model is validated with a comparative performance. However, the detection accuracy is low in this model.

ChunheSong et al. [28] suggested a feature selection-based DL model detect anomalous events. This combines the

attributes of LSTM and extreme gradient boosting topology to identify the dataset pattern. This model not only identifies the anomalous events but also prevents the false data injection attack by matching the dataset pattern. In addition, the detection accuracy of the developed model is estimated in two different ways. Hence, the system is more scalable and reliable. Moreover, Bayesian optimization is utilized to optimize the sensitive grid parameters. The integration of optimization reduces the effect of over-fitting and increases the system's efficiency. But this method does not apply to a large-scale smart grid system.

Ying Zhang et al. [29] suggested a auto-encoder-based intrusion detection scheme. In power systems, the FDI attack reduces the stability and scalability performance. This data-driven DL-based detection algorithm recognizes unobservable FDI attacks by identifying unconformity between abnormal and secure measurements. In addition, a GAN framework was deployed to capture the malicious data and neglect it. The efficiency of the developed scheme is determined by numerically simulating it in the unbalanced IEEE 123-bus and 13-bus systems. However, it does not identify the attacks other than FDI.

3. SYSTEM MODEL

The smart grid technology is electricity network which utilizes the advanced digital technologies to control and monitor the electricity transportation from all production sources to meet the energy demand. The false injected data reduces the grid performances like reliability, efficiency, and energy production. The present cyber-detection systems face challenges in detecting the all types of cyber-attacks accurately. Moreover, they require large resources to train and implement it. Developing easy and accurate cyber-attack detection mechanism is still a challenging factor in the smart grid system. Designing of cyber-attack detection model must incorporate an intelligent model with optimization technique to predict the falsely injected data effectively at cheap cost. Therefore, to overcome the challenges in the traditional system an optimized neural-based cyber-detection model was developed in this article.

4. PROPOSED ALBMNN FRAMEWORK FOR ANOMALY DETECTION

A novel hybrid Ant-Lion-based Modular Neural Network (MNN) was proposed in the particle to identify the malicious events in the large-scale smart grid. First, a sizable dataset on smart grids was collected from the standard website (Kaggle) and added to the system. Then the hybrid cyber-attack detection model was developed with the attributes of the ant-lion optimization (ALO) algorithm, and the MNN. Further, the initialized dataset is trained and pre-processed to eliminate the errors data or null values using the MNN features.

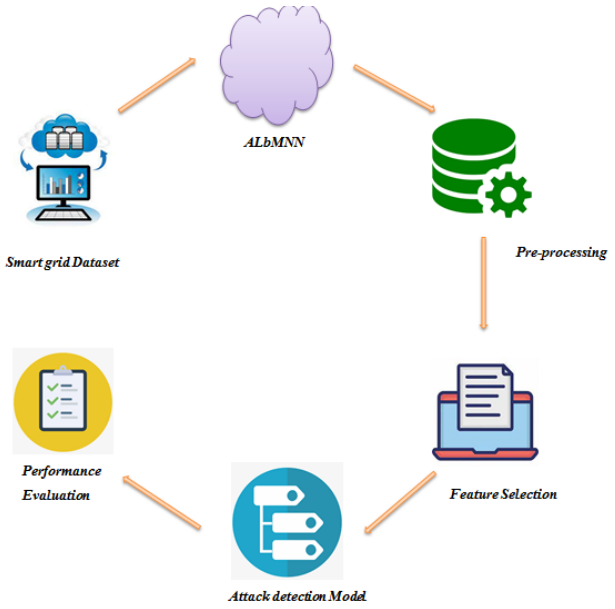


Figure 1. ALbMNN Framework

The ant-lion fitness function is used in the classification layer to choose features from the dataset. The features of the extracted data are then classified as either benign or harmful. For validation reasons, the performance of the model that is being given is also estimated and contrasted with a few conventional approaches. Fig. 1 shows the proposed methodology.

4.1. ALbMNN Layers

The input layer, hidden layer and output layer are the three layers that make up the developed ALbMNN model. The input dataset is set up and trained in the system in the input layer. Hidden Layer 1, hidden layer 2, and hidden layer 3 are the three partitions of the hidden layer. To remove incorrect data and null values, the input dataset is pre-processed and filtered in hidden layer 1.

In hidden layer 2, the dataset features are extracted using the MNN features and in hidden layer 3, the extracted features are classified as benign or malicious. The results of the model are calculated and contrasted with traditional schemes in the last layer. The layers of the suggested model are illustrated in Fig 2.

4.1.1. Data Pre-processing

The dataset contains the attributes like energy production rate, reaction time, grid stability factor, energy consumption rate, etc. To start the prediction process, the collected dataset was trained and initialized in the system. The initialization function is represented in Eqn. (1).

$$f_{in}(S_{GD}) = \{Dt_1, Dt_2, Dt_3, Dt_4, \ldots . Dt_k\} \quad (1)$$

Here, f_{in} refers to the dataset initialization function, S_{GD} denotes the large-scale smart grid dataset, f_{in} defines the data present in the dataset, and $k$indicates the total number of data

present in the dataset. pre-processing not only removes the errors but also makes the attack-detection process easy.

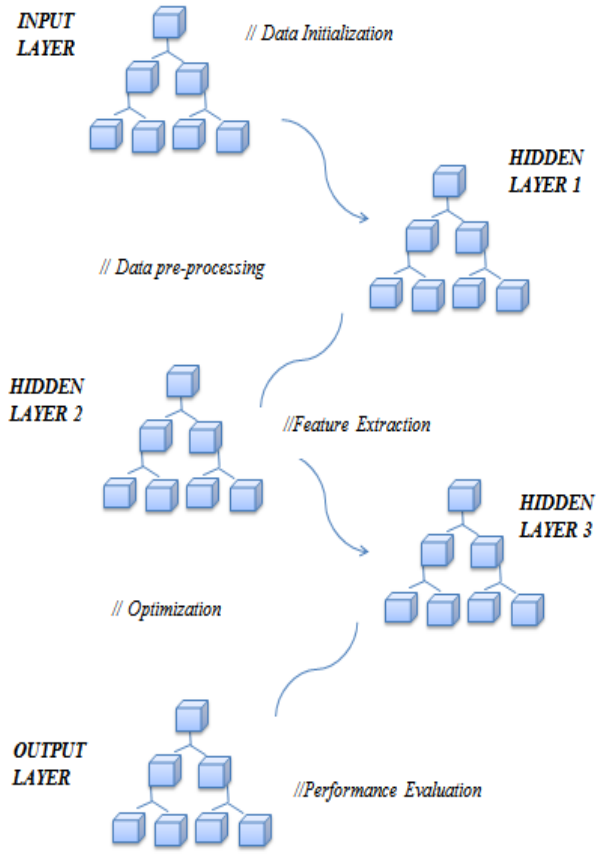


Figure 2. ALbMNN Layers

In the proposed model, the dataset was filtered using the attributes of the MNN algorithm. The key feature of the MNN algorithm is that it can process a huge dataset with less complexity. The data pre-processing function is expressed in Eqn. (2).

$$P_r^*(S_{GD}) = \sigma \sum_{i=1}^{k} (Dt_i - \delta Dt_i) \quad (2)$$

Where P_r^*represents the pre-processing function, σ indicates the pre-processing variable, andδDt_i denotes the error or null values.

4.1.2. Feature Extraction

Feature extraction is the process of selecting the important features from the pre-processed image. Both significant and insignificant structures can be found in the pre-processed dataset. But for attack detection, only meaningful features are considered. Therefore, in this stage, the pre-processed dataset's important features are retrieved, and its nonsensical features are eliminated. To extract the features, an ant-lion fitness solution is applied in the proposed model. An ALO is an algorithm based on the hunting nature of the ant lions. It involves five hunting steps namely: agent's random walk, entrapments of ants in traps, building traps, rebuilding traps, and catching prey. The ant lion fitness function is expressed in Eqn. (3).

$$F_{er}^{'}(S_{GD}) = Al_f + \frac{S_{GD} \times (Dt)}{\nu(nDt, mDt)} = ef \quad (3)$$

Where, $F_{er}^{'}$indicates the feature extraction function, Al_frepresents the ant lion fitness, nDtrefers to the normal data, mDtdenotes the malicious data, efspecifies the extracted features, and νdefines the feature tracking variable. The extracted features are further compared with the trained normal features for classification purposes.

4.1.3. Cyber-attack detection

After feature extraction, the next step is false data classification. The extracted features contain both malicious and normal data. The classification is carried out to categorize the normal and malicious data separately. The attack classification is represented in Eqn. (4).

$$A_{cl} = \begin{cases} if\left(ef = T_{nf}\right); Normal \\ if\left(ef \neq T_{nf}\right); FalseData \end{cases} \quad (4)$$

Here, A_{cl}refers to the attack classification function, and T_{nf}indicates the trained normal features. If the extracted features match with the trained normal features, it is classified as “Normal Data”. If the extracted feature does not match the trained features, it is classified as "Malicious data". Thus, the presented model detects and classifies the data as normal or malicious. Algorithm 1 provides an example of how the presented model operates.

Algorithm 1: ***Cyber-attack detection model***

Step 1: *Initialize the input smart grid dataset* S_{GD}

Step 2: *Develop the proposed model with the AOA and MNN features*

Step 3: *Pre-process the dataset to eliminate the errors*

$$P_r^* \Rightarrow \sigma(Dt_i - \delta Dt_i)$$

Step 4: *Extract the features from the dataset for classification*

$$F_{er}^{'} = Al_f + \nu(nDt, mDt) = ef$$

Step 5: *Classification of data as “Injected” or Normal*

Step 6: *Evaluate the system performance*

Step 7: *Terminate the process*

The flowchart of the developed model is displayed in Fig 3. Initially, the dataset was filtered and pre-processed to eliminate errors. Then the optimal features are extracted using the ant lion fitness function. Then the extracted data are classified as normal or malicious data by matching it with the trained normal features.

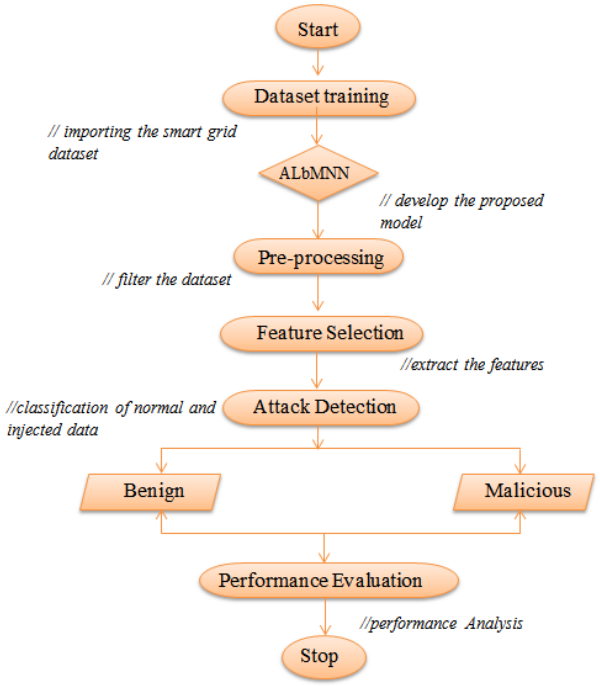


Figure 3. Flowchart of the suggested model

5. RESULTS AND DISCUSSION

A hybrid cyber-attack detection model was developed in this article to detect and classify normal and malicious data. The presented model was executed in the MATLAB software version R2020a running in windows 10. Initially, the dataset was initialized and filtered to eliminate unwanted features. Further, the features are extracted and classified as normal or false data.

Table 1. Parameters and specifications

Implementation Tools	
Parameters	**Specification**
Platform	MATLAB
Version	R2020a
OS	Windows 10
Datasets	Large-scale smart grid dataset
Application	Smart Grid

The integration of ant lion fitness in the hidden layer of the MNN offers better accuracy in detecting false data or attacks. Table 1 lists the implementation parameters and its specification.

5.1. Performance Analysis

In order to validate the suggested model, its performance is calculated and contrasted with a few traditional attack detection algorithms in this section. Performance parameters including accuracy, false positive rate and true positive rate are determined in this instance utilizing MATLAB software. Comparative analysis is conducted using the standard cyber-attack detection algorithms, such as Unsupervised ML Systems for Cyber-Attack Detection (UMLS_CD) [31], Extremely Randomized Trees-Based Schemes for Cyber-Attack Detection (ERTbS_CD) [32], Whale Optimization Algorithm-based Artificial Neural Network (WOA_ANN) [33], and Wavelet Convolutional Neural Network for Cyber Attack Detection (WCNN_CAD) [34].

5.1.1. Accuracy

The system accuracy defines the cyber-attack detection/ identification rate, i.e.) how precisely the system identifies the injected data in the smart grid. The accuracy of the presented model is formulated in Eqn. (5).

$$S_{AR} = \frac{\lambda^{+}+\lambda^{-}}{\lambda^{+}+\lambda^{-}+\alpha^{+}+\alpha^{-}} \quad (5)$$

Where, S_{AR} indicates the system accuracyλ^{+}, λ^{-} α^{+} and α^{-}denotes the true-positive, true-negative, false-positive, and false-negative, respectively.

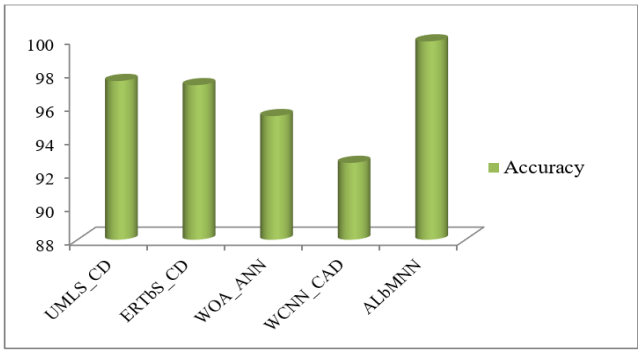


Figure 4. Comparison of Accuracy

Here, the accuracy in the existing models is calculated by executing it in the same platform for the same grid dataset. The accuracy that the existing approaches like UMLS_CD, ERTbS_CD, WOA_ANN, and WCNN_CAD attained accuracy of 97.5%, 97.25%, 95.4%, and 92.6%, respectively. Similarly, the accuracy earned by the developed model for detecting cyber-attack in large-scale datasets is 99.87%. The accuracy validation is illustrated in Fig 4.

5.1.2. True Positive Rate (TPR)

The entire positive prediction of the positive class divided by the correct prediction of malicious traits is known as the TPR. Another name for it is sensitivity. The true-positive value is divided by the true-positive and false-negative values to find it. The TPR of the system is formulated in Eqn. (6).

$$T_{PR} = \frac{\lambda^{+}}{\lambda^{+}+\alpha^{-}} \quad (6)$$

Here, T_{PR}indicates the system TPR.

The TPR percentage must be high for an effective intrusion detection mechanism. Hence, to manifest that the presented cyber-attack detection model attained higher TPR

it is compared with some existing algorithms. Here, the existing techniques like UMLS_CD, ERTbS_CD, WOA_ANN, and WCNN_CAD are implemented in the same execution platform for detecting false data injection in large-scale smart grid datasets. Following implementation, the aforementioned algorithm is used to determine the TPR from the confusion matrix.

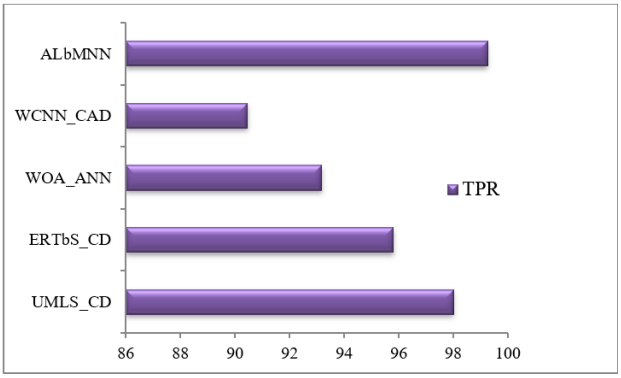


Figure 5. TPR validation

The TPR obtained by the existing algorithms is 98%, 95.8%, 93.18%, and 90.45%, respectively. However, from the comparative analysis, it is observed that the TPR achieved by the developed model is high i.e. 99.25%. The TPR comparison is shown in Fig 5.

5.1.3. False Positive Rate (FPR)

In FPR, the number of negative events is classified as a positive event. A false-positive value is definite as the ratio of false-positives to true-negatives. It is expressed in Eqn. (7).

$$F_{PR} = \frac{\alpha^{+}}{\lambda^{-}+\alpha^{+}} \tag{7}$$

WhereF_{PR} indicates the false-positive rate.

The comparison of FPR is displayed in Fig 6. FPR is one of the important parameters which determine the efficiency of the detection model.

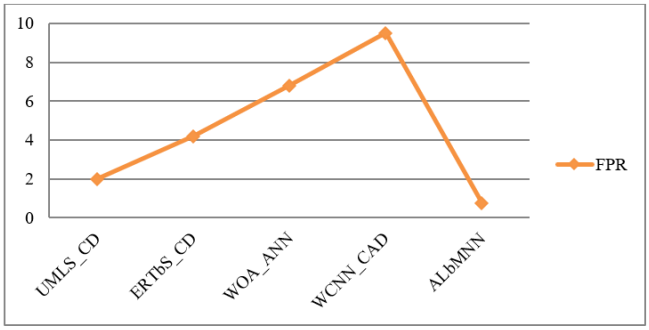


Figure 6. Comparison of FPR

The lower FPR represents the detection model correctly classifies the malicious and benign data in the dataset. Hence, the FPR of the intrusion detection model must be low. The proposed model's FPR is verified by contrasting it with the existing algorithms like UMLS_CD, ERTbS_CD, WOA_ANN, and WCNN_CAD. It is noticed that the existing techniques earned FPR of 2%, 4.2%, 6.8%, and 9.5%, respectively, whereas the proposed technique earned very FPR of 0.75%.

5.2. Discussion

An optimized neural-based intrusion detection model was presented in this paper to detect a malicious event in the smart grid system. The presented model was implemented and verified with a large-scale smart grid dataset.

Table 2. Comparative Assessment

Techniques	Accuracy (%)	TPR (%)	FPR (%)
UMLS_CD	97.5	98	2
ERTbS_CD	97.25	95.8	4.2
WOA_ANN	95.4	93.18	6.8
WCNN_CAD	92.6	90.45	9.5
ALbMNN	99.87	99.25	0.75

Finally, the outcomes of the established model were estimated and compared with different traditional schemes for validation purposes. Furthermore, the performance improvement score was determined from the comparative analysis. The overall comparative analysis was tabulated in Table 2.

6. CONCLUSION

The growth of intelligent technologies in the power grid system increases the possibility of vulnerabilities. Thus, detecting false data in smart grid systems requires an effective detection scheme. The presented detection model integrates the key features of ALO and MNN. The presented model performances are validated with a large-scale smart grid dataset. In the initial phase, the dataset was imported and trained into the system. Further, the dataset was pre-processed and the effective features are extracted to classify the normal and false data. Additionally, the parameter enhancement score was calculated by comparing the created scheme's results with those of existing methodologies. From the comparative performance of different techniques, it is observed that in the suggested model the accuracy was enhanced by 2.37%, the FPR is minimized by 1.25%, and TPR is increased by 1.25%. Thus, the developed detection scheme effectively detects and classifies the malicious data in the grid system.

CONFLICTS OF INTEREST

Not applicable.

FUNDING STATEMENT

Not applicable.

ACKNOWLEDGEMENTS

The supervisor's advice and continuous support during this research have been greatly appreciated by the author.

REFERENCES

[1] S. Ali, and Y. Li, "Learning multilevel auto-encoders for DDoS attack detection in smart grid network," *IEEE Access*, vol. 7, pp. 108647-108659, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[2] A. Sultana, A. Bardalai, and K. K. Sarma, "Salp swarm-artificial neural network based cyber-attack detection in smart grid," *Neural Processing Letters*, vol. 54, no. 4, pp. 2861-2883, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[3] S. Y. Diaba, M. Shafie-khah, and M. Elmusrati, "On the performance metrics for cyber-physical attack detection in smart grid," *Soft Computing*, vol. 26, no. 23, pp. 13109-13118, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[4] Ö. Sen, D. van der Velde, K. A. Wehrmeister, I. Hacker, M. Henze, and M. Andres, "On using contextual correlation to detect multi-stage cyber attacks in smart grids," *Sustainable Energy, Grids and Networks*, vol. 32, pp. 100821, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[5] T. Berghout, M. Benbouzid, and S. M. Muyeen, "Machine learning for cybersecurity in smart grids: A comprehensive review-based study on methods, solutions, and prospects," *International Journal of Critical Infrastructure Protection*, vol. 38, pp. 100547, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[6] J. Chen, A. J. Gallo, S. Yan, T. Parisini, and S. Y. R. Hui, "Cyber-attack detection and countermeasure for distributed electric springs for smart grid applications," *IEEE Access*, vol. 10, pp. 13182-13192, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[7] J. J. Yan, G. H. Yang, and Y. Wang, "Dynamic reduced-order observer-based detection of false data injection attacks with application to smart grid systems," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 10, pp. 6712-6722, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[8] Y. Li, X. Wei, Y. Li, Z. Dong, and M. Shahidehpour, "Detection of false data injection attacks in smart grid: A secure federated deep learning approach," *IEEE Transactions on Smart Grid*, vol. 13, no. 6, pp. 4862-4872, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[9] M. Abdel-Basset, N. Moustafa, and H. Hawash, "Privacy-preserved generative network for trustworthy anomaly detection in smart grids: A federated semisupervised approach," *IEEE transactions on industrial informatics*, vol. 19, no. 1, pp. 995-1005, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[10] R. Sharma, A. M. Joshi, C. Sahu, G. Sharma, K. T. Akindeji, and S. Sharma, "Semi supervised cyber attack detection system for smart grid," In *2022 30th Southern African Universities Power Engineering Conference (SAUPEC)*, pp. 1-5, 2022. IEEE. [CrossRef] [Google Scholar] [Publisher Link]

[11] P. K. Jena, S. Ghosh, and E. Koley, "Identification of strategic sensor locations for intrusion detection and classification in smart grid networks," *International Journal of Electrical Power & Energy Systems*, vol. 139, pp. 107970, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[12] A. N. Alkuwari, S. Al-Kuwari, and M. Qaraqe, "Anomaly detection in smart grids: a survey from cybersecurity perspective," In *2022 3rd International Conference on Smart Grid and Renewable Energy (SGRE)*, pp. 1-7, 2022. IEEE. [CrossRef] [Google Scholar] [Publisher Link]

[13] Y. Li, and J. Yan, "Cybersecurity of smart inverters in the smart grid: A survey," *IEEE Transactions on Power Electronics*, vol. 38, no. 2, pp. 2364-2383, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[14] O. Boyaci, M. R. Narimani, K. Davis, and E. Serpedin, "Cyberattack detection in large-scale smart grids using chebyshev graph convolutional networks," In *2022 9th International Conference on Electrical and Electronics Engineering (ICEEE)*, pp. 217-221, 2022. IEEE. [CrossRef] [Google Scholar] [Publisher Link]

[15] D. An, F. Zhang, Q. Yang, and C. Zhang, "Data integrity attack in dynamic state estimation of smart grid: Attack model and countermeasures," *IEEE Transactions on Automation Science and Engineering*, vol. 19, no. 3, pp. 1631-1644, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[16] S. Sharda, K. Sharma, and M. Singh, "False Data Injection and Detection in Smart Grid Cyber-Physical Systems by Iterative Load Flow Analysis," In *Advances in Information Communication Technology and Computing: Proceedings of AICTC 2021*, pp. 245-257, 2022, Singapore: Springer Nature Singapore. [CrossRef] [Google Scholar] [Publisher Link]

[17] Q. Li, J. Zhang, J. Zhao, J. Ye, W. Song, and F. Li, "Adaptive hierarchical cyber attack detection and localization in active distribution systems," *IEEE transactions on smart grid*, vol. 13, no. 3, pp. 2369-2380 2022. [CrossRef] [Google Scholar] [Publisher Link]

[18] T. Teng, and L. Ma, "Deep learning-based risk management of financial market in smart grid," *Computers and Electrical Engineering*, vol. 99, pp. 107844, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[19] Y. Liu, and L. Cheng, "Relentless false data injection attacks against Kalman-filter-based detection in smart grid," *IEEE Transactions on Control of Network Systems*, vol. 9, no. 3, pp.1238-1250, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[20] H. T. Reda, A. Anwar, A. Mahmood, and N. Chilamkurti, "Data-driven approach for state prediction and detection of false data injection attacks in smart grid," *Journal of Modern Power Systems and Clean Energy*, vol. 11, no. 2, pp. 455-467, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[21] J. Shi, S. Liu, B. Chen, and L. Yu, "Distributed data-driven intrusion detection for sparse stealthy FDI attacks in smart grids," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 68, no. 3, pp. 993-997, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[22] D. Mukherjee, S. Chakraborty, A. Y. Abdelaziz, and A. El-Shahat, "Deep learning-based identification of false data injection attacks on modern smart grids," *Energy Reports*, vol. 8, pp. 919-930, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[23] S. H. Majidi, S. Hadayeghparast, and H. Karimipour, "FDI attack detection using extra trees algorithm and deep learning algorithm-autoencoder in smart grid," *International Journal of Critical Infrastructure Protection*, vol. 37, pp. 100508, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[24] S. H. Majidi, S. Hadayeghparast, and H. Karimipour, "FDI attack detection using extra trees algorithm and deep learning algorithm-autoencoder in smart grid," *International Journal of Critical Infrastructure Protection*, vol. 37, pp. 100508, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[25] S. Sengan, V. Subramaniyaswamy, V. Indragandhi, P. Velayutham, and L. Ravi, "Detection of false data cyber-attacks for the assessment of security in smart grid using deep learning," *Computers & Electrical Engineering*, vol. 93, pp.107211, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[26] M. R. C. Acosta, S. Ahmed, C. E. Garcia, and I. Koo, "Extremely randomized trees-based scheme for stealthy cyber-attack detection in smart grid networks," *IEEE access*, vol. 8, pp.19921-19933, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[27] Y. Tian, Q. Wang, Z. Guo, H. Zhao, S. Khan, W. Mao, and J. Zhao, "A hybrid deep learning and ensemble learning mechanism for damaged power line detection in smart grids," *Soft Computing*, vol. 26, no. 20, pp.10553-10561, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[28] C. Song, Y. Sun, G. Han, and J. J. Rodrigues, "Intrusion detection based on hybrid classifiers for smart grid," *Computers & Electrical Engineering*, vol. 93, pp. 107212, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[29] Y. Zhang, J. Wang, and B. Chen, "Detecting false data injection attacks in smart grids: A semi-supervised deep learning approach," *IEEE Transactions on Smart Grid*, vol. 12, no. 1, pp. 623-634, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[30] L. Abualigah, and A. Diabat, "A novel hybrid antlion optimization algorithm for multi-objective task scheduling problems in cloud computing environments," *Cluster Computing*, vol. 24, no. 1, pp. 205-223, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[31] S. Varela-Santos, and P. Melin, "A new modular neural network approach with fuzzy response integration for lung disease classification based on multiple objective feature optimization in chest X-ray images," *Expert Systems with Applications*, vol. 168, pp. 114361, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[32] H. Karimipour, A. Dehghantanha, R. M. Parizi, K. K. R. Choo, and H. Leung, "A deep and scalable unsupervised machine learning system for cyber-attack detection in large-scale smart grids," *Ieee Access*, vol. 7, pp. 80778-80788, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[33] L. Haghnegahdar, and Y. Wang, "A whale optimization algorithm-trained artificial neural network for smart grid cyber intrusion detection," *Neural computing and applications*, vol. 32, no. 13, pp. 9427-9441, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[34] J. P. Li, G. U. Nneji, A. Z. Yutra, B. D. Lemessa, S. Nahar, E. C. James, and A. U. Haq, "The Capability of Wavelet Convolutional Neural Network for Detecting Cyber Attack of Distributed Denial of Service in Smart Grid," In *2021 18th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP)*, pp. 413-418, 2021. IEEE. [CrossRef] [Google Scholar] [Publisher Link]

AUTHORS

R.A. Mabel Rose received her B.E. degree and M.E. degree in Computer Science and Engineering from Anna University, Chennai, India. She started her career as Lecturer and has 13 years of experience. Currently she is working as Assistant Professor in Panimalar Engineering College, Chennai. Her research interests include Cyber Security and Cloud Computing. She is a lifetime member of ISTE.

V. Padmajothi Obtained Ph.D. at Veltech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology in the year 2023, M.E. (Applied Electronics) in Arulmigu Meenakshi Ammal College of Engineering (ANNA University) in the year 2005, Obtained B.E. (EEE) in Adhiparasakthi College of Engineering (Madras University). Currently working as Assistant professor in SRM Institute of Science and Technology, Kattankulathur, Chennai from the year 2005. Having 19 years of teaching experience and teaching various courses such as Embedded Linux, Hardware Interfacing and Networking, Embedded C, Embedded System Design using Raspberry Pi, Embedded System Design using Arduino, Real Time Systems, Control Systems to B.Tech students. Areas of interest Embedded Control Systems, Cyber Physical Systems, Deep Learning, Machine Learning.

Arrived: 27.09.2024
Accepted: 11.10.2024