RESEARCH PAPER

A Quantum-Optimized Fuzzy Min-Max Neural Network for Securing Wireless Sensor Networks

[1] Dr. D. Princy, [2] Mrs. G. K. Karthika, [3] Mrs. C. Mercy Praba, [4] Dr. S. Devibala

[1]Assistant Professor, Dept. of Computer Science, Dr. SNS Rajalakshmi College of Arts and Science (Autonomous), Coimbatore - 641 049, India. Email: princyd287@gmail.com. https://orcid.org/0000-0003-2691-4794

[2]Assistant Professor, PG & Research Dept. of Computer Science, Sri Ramakrishna College of Arts & Science (Autonomous), Coimbatore - 641 006, India. Email: karthiguna1011@gmail.com

[3]Assistant Professor, Dept. of Computer Science, Dr. N.G.P. Arts and Science College, Coimbatore, India. Email: mercy.ccf5@gmail.com

[4]Assistant Professor, PG & Research Dept. of Computer Science, Sri Ramakrishna College of Arts & Science (Autonomous), Coimbatore - 641 006, India. Email: shiiv30@gmail.com

Abstract: Modern cyber-physical systems use Wireless Sensor Networks (WSNs), which are in principle susceptible to a large variety of cyberattacks because they have a decentralized structure and limited resources. In this paper a new intrusion detection system inspired by quantum optimization, based on the Fuzzy Min-Max Neural Network (FMNN) and quantum-inspired optimization to detect threats efficiently and robustly. The FMNN architecture supports dynamic attention multivariate sequence with the ability to model contextual dependencies very finely in network traffic. Quantum Particle Swarm Optimization (QPSO) is implemented to solve the problem of the large dimension search due to model tuning that guarantees a global optimization offering global conversing along with less computation. The empirical testing test on the network simulation indicates that CNN–QPSO–FMMNN achieves detection accuracy in training time relative to the classical IDS models. The proposed framework has shown a strong robustness to detect known and zero-day attacks as well as being computationally efficient and thus it is ideally fit in an advanced networking system to provide real time security. The work demonstrates itself in the synergy between quantum-inspired optimization and neural network based deep architectures in state-of-the-art in intelligent network security.

Keywords: Security, intrusion, quantum computing, particle swarm, neural network, and accuracy.

How to cite this article: Princy D, Karthika GK, Praba CM, Devibala S. A Quantum-Optimized Fuzzy Min-Max Neural Network for Securing Wireless Sensor Networks. Int J Drug Deliv Technol. 2026;16(10s): 860-869; DOI: 10.25258/ijddt.16.10s.101

1. Introduction

The progression accomplished on "green IT," numerous organizations initiated to identify the approaches to defeat the economy's stagnation and reduce the IT cost. Cloud computing is an emerging technology where payment is made based on service utilization without purchasing expensive physical equipment [1]. Cloud computing offers convenient, on-demand, and ubiquitous access to networks and services, determined as a pay-per-use-basis service for the customer. It is identified that the cloud delivers its service through the internet, anywhere and anytime. In contrast, a pool of configurable resources for computing is provided (e.g., storage, network, services, servers, and application) [2].

The cloud service is accessed via the internet where the connection and interface's namely tablet, laptop, smartphones, and computer, are necessary for utilizing the services. The scalability of cloud computing is attained by visualization technology. The concept of grid computing has emerged from cloud computing, where large-scale and complex mathematical issues are solved [3]. The paradigm of cloud computing is a prominent framework that is contributed to cloud evolution in diverse aspects. Cloud computing delivers simple management of servers, resource utilization, minimization of cost, and consolidation of servers [4].

Besides the advantages of cloud computing, diverse security threats and attacks are a considerable obstacle. It prevents an administration from transmitting their business to the cloud [5]. In the distributed context of cloud computing, multi-tenant utilization of virtualization, and their dependence on the internet, attackers exploit several vulnerabilities to security to undermine the integrity, availability, and confidentiality of the cloud services and resources [6]. One of the significant threats in cloud computing is

insider attack, where the virtual machines (VM) are susceptible to attack by the infrastructure or host of the physical framework that prevents delivering the needed services [7]. For instance, the malware on one VM in the cloud network can be installed in the client of other VM's. Then, the system is utilized as a zombie machine to initiate the Distributed Denial of Service (DDoS) attack against additional VMs in a similar infrastructure [8].

In the current technology scenario, insider attack poses complicated challenges for the cloud computing where the connectivity and service unavailability issue is deactivated by the full service [9]. The behavior of malicious software in the VM has to be monitored to expose the malicious actions in the cloud environment virtually. This malware can influence both the environment and operations of cloud computing. The applications and information in the cloud and VM will be affected by the malware [10]. The security feature in the cloud computing environment is successfully installed and defended by the IDS mechanism, which is an indispensable aspect of ensuring the security standards in the cloud [11].

Traditional IDS are not appropriate for the cloud environment where the VM are dynamic, and IDS installation is complicated [12]. Most of the behavioral IDS techniques are affected by the unavailability and training issues of datasets reflected in the virtual cloud evolution. This process necessitated an effective identification system to work proficiently with the virtual cloud. The data is gathered from the cloud environment to investigate the cloud-based IDS, and the attacks in the cloud are effectively classified [13]. The communication among the VM's over the cloud is monitored by the VM monitor or hypervisor [14]. The anomaly identification system is instilled at the hypervisor layer of the cloud, and practical soft computing is necessary to identify anomaly IDS in the cloud environment.

In the research work, the cloud intrusion dataset is taken into consideration to incorporate the classification and the process is achieved locally. This technique uses the NSL KDD and CICIDS2017 dataset, and categorizes incidence of attacks. This research work identifies the frequency of Normal, back, Smurf, Pod, Neptune and Teardrop. The suggested method addresses such the shortcomings in the training and classification. Also, the successful method of feature selection enhances the accuracy of classification. The developed CNN-QPSO-FMMNN is set against the current popular classification methods. The suggested system was tremendously flexible and was exploited in myriads of different applications.

Intrusion Detection Systems (IDS) play a pivoting role in achieving the security and integrity of network infrastructures by identifying malicious activities, displaying patterns of data traffic abnormalities. It can be argued that with the growing complexity of hacking attacks, the contemporary IDS systems need more intelligent computational algorithms that possess the functionality of inferring adaptively and also accurately discriminating between normal and abnormal patterns. An example of this is the Convolutional Neural Network -Quantum-behaved Particle Swarm Optimization -Fuzzy Min -Max Neural Network or CNN -QPSO -FMMNN model that combines deep learning, swarm intelligence, and fuzzy logic to effectively categorize the anomaly when defining anomaly detection tasks.

CNN is used in this hybrid framework in automatic feature extraction (spatial and local dependencies in feature maps extraction) in raw or preprocessed network traffic data. QPSO is an optimization tool that uses adaptation to optimize the parameters of a following FMMNN classifier and be able to accelerate it, as well as prevent local optima. FMMNN component It is implemented on the basis of fuzzy hyper box theory and assigns patterns into classes by forming geometric hyper boxes in feature space of many dimensions, with a hyper box labeling a particular class.

The classification process involves checking whether an incoming data instance fits within the inclusion criteria of an existing hyper box. If it does, the hyper box’s minimum and maximum boundary values are updated; if not, a new hyper box is created to represent the new pattern. This adaptability allows the system to dynamically learn emerging attack signatures without retraining from scratch. The use of fuzzy boundaries ensures that the model can handle uncertainty, overlapping patterns, and noisy data—common in real-world intrusion scenarios.

By integrating CNN’s high-level feature extraction, QPSO’s global search and optimization capabilities, and FMMNN’s interpretable fuzzy classification, CNN–QPSO–FMMNN offers an effective, scalable, and adaptive IDS solution. This synergy not only boosts detection accuracy and reduces false positives but also enhances robustness in detecting novel and sophisticated cyber threats in real time.

The remainder of the research work is arranged as: different techniques in intrusion detection and attack prediction system is discussed in Section 2, the

proposed feature selection and CNN–QPSO–FMMNN technique are detailed in Section 3, and the outcome of the proposed approach is compared with existing strategy in Section 4, and the Section 5 is concluded with future research direction.

2. Related Works

With the exponential growth of interconnected devices, particularly in the domains of IoT, IIoT, and vehicular networks, network infrastructures have become increasingly vulnerable to sophisticated cyber threats. Intrusion Detection Systems (IDS) is a critical defense mechanism for real-time threat identification and mitigation. Traditional IDS frameworks, primarily signature-based or anomaly-based, often struggle to detect zero-day attacks and adapt to dynamic network traffic patterns. The integration of artificial intelligence (AI) and machine learning (ML) into IDS design has significantly improved detection capabilities, adaptability, and automation.

Recently, some articles have examined different AI-based IDS models that focus on numerous application sceneries, such as the industrial control system, vehicle networks, and the IoT environment. The strategies are based on the use of supervised, unsupervised and a combination of patterns recognition and anomaly detection. Though impressive progress has already been made, problems, like a high cost of computations, the lack of scalability, dependency on a dataset, and the impossibility of deployment in real-time still exist. Table 1 lists outstanding recent IDS contributions, and critiquing them on their goals, methods, results and weaknesses.

Table 1. Comprehensive Analysis of IDS

Author(s) & Year	**Objective**	**Proposed Method / Approach**	**Key Findings**	**Limitations / Future Scope**
Kavitha, D., & Thejas, S. (2024) [15]	To enhance cyber threat detection through AI-based solutions for adaptive and proactive defense.	Hybrid AI models combining supervised and unsupervised ML for real-time anomaly detection.	Achieved higher detection rates and reduced false positives in diverse cyberattack scenarios.	Requires large-scale labeled datasets; computational cost optimization suggested.
Idouglid, L., Tkatek, S., Elfayq, K., & Guezzaz, A. (2024) [16]	To fortify IIoT systems using ML-powered IDS for Industry 4.0 networks.	ML algorithms (RF, SVM, KNN) integrated with IIoT-specific security frameworks.	Improved resilience against known and zero-day attacks in IIoT networks.	Limited evaluation on large-scale industrial datasets; scalability concerns remain.
Rao, D. D., et al. (2024) [17]	To strengthen IoT network layer security using advanced IDS and AI-based analytics.	AI-driven IDS with traffic pattern analysis and dynamic signature updates.	Significantly reduced attack detection latency and improved threat classification accuracy.	Need for integration with blockchain-based authentication for enhanced trust.
Kim, H. R., & Song, H. M. (2024) [18]	To develop a lightweight IDS for CAN bus security in vehicles.	Word embedding-based feature representation with lightweight classifiers.	Achieved low latency and high detection accuracy with minimal resource usage.	Requires robustness testing in real-world vehicular environments with mixed traffic.
Farrukh, Y. A., et al. (2024) [19]	To design a self-adaptive IDS that operate	Adaptive immune system-inspired IDS with	Self-sustaining threat mitigation with minimal	Future work to include multi-agent IDS collabor

	s with minimal human intervention.	self-learning capabilities.	false alarms in dynamic network conditions.	ation for distributed environments.
Ahmed, M. A. O., et al. (2025) [20]	To improve IoT IDS performance using gradient boosting techniques.	Performance Gradient Boosting (PGB) for feature selection and intrusion classification.	Outperformed conventional ML models in accuracy and detection rate on IoT datasets.	Needs real-time deployment validation and energy efficiency assessment.
Hizal, S., Cavusoglu, U., & Akgun, D. (2024) [21]	To counter IoT-based DDoS attacks using deep learning.	CNN-BiLSTM hybrid network for temporal-spatial feature learning.	High detection accuracy for volumetric and application-layer DDoS attacks.	Model complexity may hinder deployment on low-power IoT devices.

The literature review identifies the ongoing development of the Intrusion Detection System (IDS) systems; nevertheless, some of the critical issues are still not reflected. There remains limited flexibility, because many models perform well with certain datasets but do not generalize well in heterogeneous and real-time traffic. Computational complexity is another urgent issue, where deep learning and hybrid networks- despite being accurate- use so much processing power and, as such, limit deployment on resource-limited devices. Feature selection can be inefficient where optimization techniques used like Particle Swarm Optimization (PSO) tend to fail in terms of premature convergence to sub-optimal solutions in the high dimensional feature space.

Intrusion Detection Systems (IDS) are becoming important elements of contemporary cybersecurity systems because they offer preventive protections to identify, examine, and stop unauthorized operations on networks. The growing scale and sophistication of cyber threat have been the cause of increasing inadequacy of traditional rule based approaches to IDS, which clearly lack the adaptability, false positives and generalisation to new attack patterns (Buczak & Guven, 2016). Recent trends in research have therefore turned to machine learning (ML) and deep learning (DL) methods deployed utilizing their ability to produce models of the complex, non-linear correlation between network traffic data to increase performance in detecting the anomalies (Khan et al., 2019).

Convolutional Neural Networks (CNNs) are one of such and have proven their proficiency in extraction of features in high dimensional data, by considering spatial dependencies and could minimize the need of manual preprocessing. The CNN architectures have some very beneficial features in IDS applications since they can learn hierarchical feature representations requiring no or minimal pre-processing of network flow data in an automated manner (Zhang et al., 2021). Nevertheless, in order to improve the convergence rate, generalization ability, and reliability of classification, CNNs may frequently need optimization schemes.

The combination of swarm intelligence algorithms and IDS has a lot of promise and is an optimizing concept currently under development Introducing swarm intelligence, including the integration with Quantum-behaved Particle Swarm Optimization (QPSO), to efficient parameter tuning by exploring the solution space is more efficient as compared to the classical gradient-based methods (Sun et al., 2012). The concept of quantum mechanics is applied to the particle swarm framework so that a broader search space can be explored by the particles in the sense that it is highly probable, but there is no guarantee that it does not lead to premature convergence as is often the case with classical Particle Swarm Optimization (PSO) and Reply.

CNN plus QPSO and Fuzzy Neural Networks Min-Max (FMMNN) CNN CNN and QPSO The CNN QPSO FMMNN model integrates the CNN together with QPSO and Fuzzy Min-Max Neural Networks (FMMNN). Classification of the FMMNN component is an essentially interpretable classification mechanism with hyper boxes in the feature space which is defined by minimum and maximum points and provides an easy way of dealing with uncertainty and overlapping boundaries of classes (Simpson, 1992). The incorporation of QPSO guarantees that parameters of the CNN will be optimally tuned, whereas the FMMNN classifier increases the level of decision explanability and readiness to various attack patterns.

The above categories of hybrid architectures have been shown to increase classification accuracy, precision, recall and F1- score by a large percentage than standalone CNN or conventional IDS models (Zhou et al., 2023). This state is especially applicable in identification of rare but high-impact intrusions in which the detection rate ought to remain high without triggering excessive false positives to avoid the security degradation of operations. As the volume, velocity, and variety of traffic across networks continue to grow in cloud-based and IoT-enabled networks, frameworks such as CNNQPSOFMMNN may be an important leap to next-generation IDS that optimize detection performance with computational efficiency and interpretability.

This paper tests the CNNQPSOFMMNN model against benchmark datasets (intrusion detection) with a standard set of classification measures, such as Accuracy, Precision, Recall, F1-score, and Detection Rate. This research helps build the body of knowledge in the field of hybrid deep learning and swarm intelligence-based sources of cybersecurity by researching its competence accordingly to the existing baseline and state-of-the-art methods of IDS.

Emerging threats detection, especially advanced/stealth zero-day attacks are not optimally detected. Real-life applicability is also compounded by scalability problems, and little testing on large, multi-domain datasets casts doubts on applicability, operationally. With the aim of overcoming such drawbacks, the current study proposes an IDS based on CNN-QPSO-FMMNN, with deep convolutional feature extraction, the reinforcement of the feature selection method using quantum-inspired particle swarms, and fuzzy nonlinear classification using the min-max neural network. The hybridised model is aimed at increasing detection accuracy, scalability and reducing false positives and computational expenses to enable effective use in diverse and dynamic network environments.

3. Proposed Methodology - CNN–QPSO–FMMNN Based Intrusion Detection Framework

The section describes proposed anomaly detection methodology that has a consistent flow of stages. Raw input data is first cleaned up by preprocessing procedures which remove noise and normalize formats so that data can be analysed next. The cleaned data is then handed over to the stages of feature selection, where irrelevant or redundant attributes will be deleted in order to increase efficiency and precision. The CNN–QPSO–FMMNN classification technique is applied to selected properties and distinguished normal and abnormal patterns in an ample way.

1.1. Pre-Processing

The dataset is first transformed by converting symbolic protocol names {TCP, UDP, ICM} into numeric values {1, 2, 3} using a representation method [22]. Following this, feature values are normalized using the formula:

$$f = \frac{f - minm}{maxim - minm}$$

where f is the feature, minm is the minimum, and maxim is the maximum value. This normalization scales all features to the range (0, 1).

1.2. QPSO for Feature Selection

Feature selection is a significant concern in the detection of intrusion in the cloud environment. There are several features in the intrusion detection dataset, and the purpose of intrusion is monitored. In the dataset, some of the features are significant, and some of them are not necessary. Hence feature selection process is accomplished by a convolutional neural network (CNN) [23]. The lessening of insignificant features enhances the detection accuracy, eventually boosting the computation process, thus enriching the entire performance of IDS. In a specific context, where there are no impractical features, the performance scale of IDS can be improved by focusing on essential attributes without influencing the accurateness of identification.

To further improve the dimensionality reduction and to ensure the optimal subset of features is selected, a hybrid feature selection strategy is employed that integrates CNN with Quantum-inspired Particle Swarm Optimization (QPSO). In this dual-stage mechanism, CNN serves as a filter-based selector to rank initial features based on feature importance and activation impact, while QPSO acts as a wrapper-based optimizer. The QPSO algorithm operates in a reduced search space created by CNN, leveraging quantum behavior to balance exploration and exploitation. Each particle in QPSO represents a binary string of selected features, and its fitness is evaluated using classification accuracy via a validation set. This synergy enables the elimination of redundant and noisy attributes while retaining semantically rich features, enhancing classifier performance. In this approach, 41 features are retrieved from the NSL-KDD dataset, further transmitted to the classification phase.

Algorithm 1. QPSO-Based Feature Optimization
Initialize particle positions and velocities in the reduced feature space
Set the personal best and global best fitness

```
While stopping criteria not met do
  For each particle
    Evaluate fitness using classification accuracy
    Update personal and global best
    Update position using QPSO position equation:
      x_i(t+1) = p_i ± β|mbest - x_i| * ln(1/u), u ∈ (0,1)
  End for
End while
Return best feature subset configuration
```

1.3. Anomaly Detection

The proposed classification framework grounded in fuzzy set theory within hyperbox structures. The network architecture comprises NN input nodes, HH hyperboxes in the hidden layer, and LL output nodes corresponding to labeled classes. Each network input is connected to every hyperbox, and each hyperbox is associated with a specific class label. A hyperbox represents a defined region in nn-dimensional space that encapsulates a set of input patterns with complete class membership.

The hyperbox boundaries are defined in terms of minimum and maximum points and its classification pattern is determined by a membership. This membership is a number that measures the extent to which a given fuzzy set has more belongingness to a particular input pattern quantified within the hyperbox, where averages between 0 (indicating that its set has no membership) and 1 (full membership). The learning mechanism in FMMNN works in three basic steps, expansion where the hyperboxes are modified to incorporate new trends, contraction where the overlaps between hyperboxes belonging to different classes are minimized and testing where the classification rate is evaluated. In this manner, FMMNN is capable of managing issues of classification and data overlap efficiently, boasting versatility and competence across uncertain or noisy settings in data.

1.3.1. Fuzzy set hyper box

A fuzzy set hyper box implies a set of VMs in the n dimensional space and pattern of information. The area of a hyper box is identified using the minimum point, maximum point and membership function. It encompasses the information that has the highest membership function, and the hyper box set is defined as follows,

$$V_m = \{U, p_m, q_m, Z(U, p_m, q_m)\} \forall U \in L^n$$

where Vm indicates the jth hyper box, vector or pattern in the input with n-dimension is indicated as U, minimum point is denoted as p_m, the max point is indicated as q_m, and the Z(U, p_m, q_m) is the membership function. The anomaly data pattern U(u_1, u_2, u_3, ….. u_n,) is determined as follows,

$$p_m = (p_{m1}, p_{m2}, p_{m3}, \dots\dots\dots, p_{mn})$$
$$q_m = (q_{m1}, q_{m2}, q_{m3}, \dots\dots\dots, q_{mn})$$

The hyper box is determined from the above equation, and accumulated data is concluded with k_{th} pattern of attacks in W_k ,which is determined as the union set of hyper boxes that is K. This set of classes rely on the k^{th} pattern class and is defined as follows,

$$W_k = \cup_{m \in K} V_m$$

The fuzzy membership function detects the normal and the degree of attack from the input in the hyper box. The range of input patter is determined as A_m that lies among the values 0 and 1. The value 1 in the hyper box denotes the full membership function. The sum of two's complement values is the average of min and max point violation. The membership function is determined as follows,

$$B_m(D_h) = \frac{1}{2n}\sum_{l=1}^{n}\Big[\max\big(0,1 - \max\big(0,\gamma\,\min\big(1, d_{hl} - q_{jl}\big)\big)\big) + \max i\,(0,1 - \max i(0,\gamma\,\min(1, v_{ml} - d_{hl}))\Big]$$

In FMMNN, anomaly patterns are represented by $B_m(D_h)$ that is the membership function, corresponding to a hyperbox. The sensitivity parameter γ controls the decline in membership value as the distance among the input pattern $D_h = (D_{h1}, D_{h2}, D_{h3}, \dots\dots\dots, D_{hn})$ and the hyperbox increases. The hyperbox boundaries are defined by the minimum point $p_m = (p_{m1}, p_{m2}, p_{m3}, \dots\dots\dots, p_{mn})$ and maximum point $q_m = (q_{m1}, q_{m2}, q_{m3}, \dots\dots\dots, q_{mn})$.

The hyperbox learning in IDS involves three main stages: data expansion, overlap testing, and contraction. Using the two different IDS dataset, anomaly data is structured with n attributes and one decision attribute. Initially, an input pattern with its class label is selected, and a matching hyperbox for the same class is identified. When the input is already in the hyperbox, the minimum and maximum points are updated, otherwise a new hyperbox with a set of min max points is built. Existence of hyperboxes may overlap with growing hyperbox causing overlaps hence the overlaps testing phase will pick it. To retain the class separation, contraction process is used to ensure that there are no overlaps without missing the learned patterns. This form of structure will not associate the anomaly patterns accurately and thus maintain the boundaries of classes, a factor that qualifies it during the intrusion detection process within the IDS framework. This section depicts the information about

the specifics of each sub-process and its contribution to the functioning of IDS.

The enlargement of hyperbox is aimed at the assignment of an input pattern to a current hyperbox with high membership value. In case there is no proper hyperbox, yet a new one is developed, and a new class label is assigned. The size of hyperbox is based on the threshold parameter of which 0 to 1 is included. Each hyperbox is defined for an ordered pair (D_h, w_h), where $D_h = (D_{h1}, D_{h2}, D_{h3}, \dots\dots\dots, D_{hn})$ represents the attribute set for nn inputs, and whw_h denotes the class label. The inclusion of an input pattern into a hyperbox occurs when it satisfies the membership and boundary conditions based on θ. This process ensures optimal utilization of existing hyperboxes, minimizes unnecessary creation of new ones, and maintains accurate class representation, thereby improving classification efficiency within the IDS framework.

$$n\theta \geq \sum_{l=1}^{n} (\max(q_{ml}, d_{hl})) - \min(p_{ml}, d_{hl}))$$

When inclusion criteria are met, the fuzzy hyper box updates its minimum and maximum values. These values are then computed using below formula,

$$q_{jl}^{new} = \text{mini}\left(v_{jl}^{old}, d_{hl}\right) \forall l = 1,2,3, \dots\dots, n$$

$$w_{jl}^{new} = \text{maxi}\left(w_{jl}^{old}, d_{hl}\right) \forall l = 1,2,3, \dots\dots, n$$

The main intent is to identify the overlap among the hyper box that is identical or diverse. Suppose the hyper box is expanded in the preceding step, and it denotes another class of data. If the value of $\delta^{old} - \delta^{new}$ is greater than zero, then Δ = 1 and $\delta^{old} = \delta^{new}$. The dimensional index is denoted by Δ , and if the overlap is identified, then the overlap test is carried over the entire dimension of data. If there is minimum overlap, then the contraction process is not necessary, that is Δ = -1. Initially, the δ^{old} is assigned with one. The overlap among the hyper box is tested by the cases as follows,

Case 1: $p_{jl} < p_{kl} < q_{jl} < q_{kl}$

$$\delta^{new} = \min(q_{jl} - p_{kl}, \delta^{old})$$

Case 2: $p_{kl} < p_{jl} < q_{kl} < q_{jl}$

$$\delta^{new} = \min(q_{kl} - p_{jl}, \delta^{old})$$

Case 3: $p_{jl} < p_{kl} < q_{jl} < q_{kl}$

$$\delta^{new} = \min(\min(q_{kl} - p_{lj}, q_{jl} - p_{kl}), \delta^{old})$$

Case 4: $p_{kl} < p_{jl} < q_{kl} < q_{jl}$

$$\delta^{new} = \min(\min(q_{lj} - p_{kl}, q_{kl} - p_{jl}), \delta^{old})$$

The process of contraction disturbs the hyper box-sized, and it can be minimized, which is responsible for dimension alteration and removing of overlapping's. This process relies on the value of Δ and if the Δ is greater than 0 then the value is altered, which can minimize the disturbance in the classification. The following cases generate the process of contraction,

Case 1: $p_{j\Delta} < p_{k\Delta} < q_{j\Delta} < q_{k\Delta}$

$$q_{j\Delta}^{new} = p_{k\Delta}^{new} = \frac{(q_{m\Delta}^{old} + p_{k\Delta}^{old})}{2}$$

Case 2: $p_{k\Delta} < p_{m\Delta} < q_{k\Delta} < q_{j\Delta}$

$$q_{k\Delta}^{new} = p_{k\Delta}^{new} = \frac{(q_{k\Delta}^{old} + p_{j\Delta}^{old})}{2}$$

Case 3: (a) $p_{j\Delta} < p_{k\Delta} < q_{k\Delta} < q_{j\Delta}\ and\ (q_{k\Delta} - p_{j\Delta}) < (q_{j\Delta} - p_{k\Delta})$

$$\left(q_{j\Delta}^{new} = p_{k\Delta}^{old}\right)$$

(b) $p_{j\Delta} < p_{k\Delta} < q_{k\Delta} < q_{j\Delta}\ and\ (q_{k\Delta} - p_{j\Delta}) > (q_{j\Delta} - p_{k\Delta})$

$$\left(q_{j\Delta}^{new} = p_{k\Delta}^{old}\right)$$

Case 4: (a) $p_{k\Delta} < p_{j\Delta} < q_{j\Delta} < q_{k\Delta}\ and\ (q_{k\Delta} - p_{j\Delta}) < (q_{j\Delta} - p_{k\Delta})$

$$\left(q_{k\Delta}^{new} = p_{j\Delta}^{old}\right)$$

(b) $p_{k\Delta} < p_{j\Delta} < q_{j\Delta} < q_{k\Delta}\ and\ (q_{k\Delta} - p_{j\Delta}) > (q_{j\Delta} - p_{k\Delta})$

$$\left(q_{k\Delta}^{new} = p_{j\Delta}^{old}\right)$$

The classification process is efficiently executed, enabling accurate detection of attacks within the anomaly dataset. By leveraging the trained model's capability to differentiate between normal and abnormal patterns, each incoming instance is evaluated against the learned decision boundaries. Suspicious patterns exhibiting deviations from normal behavior are flagged as potential attacks, while legitimate activities are correctly classified as normal. This precise identification minimizes false positives and false negatives, ensuring reliable anomaly detection and enhancing the overall security performance of the system.

4. Result and Discussion

The simulation was implemented in Python 3.8 using TensorFlow 2.x, Scikit-learn, and NumPy libraries on a workstation with an Intel Core i9-12900K CPU, 64 GB DDR5 RAM, and NVIDIA RTX 3090 GPU (24 GB VRAM). It used two benchmark intrusion detection datasets, NSL-KDD and UNSW-NB15. Pre-processing entailed categorical encoding, symbolic to numeric and min max normalization. A hybrid CNN-QPSO framework was used to carry out feature selection, and each feature was classified by Fuzzy Min-Max Neural Network (FMMNN). The evaluation metrics were accuracy, precision, recall, F1-score and detection rate. To permit generalisation, ten-fold cross-validation was applied and hyperparameters optimised by grid search in order to obtain maximum performance with the

respective configuration. Table 1 shows the description of network security dataset.

Table 1. Dataset Description

Dataset	No. of Records	No. of Features	Feature Types	Attack Categories	Normal Records	Attack Records	Year of Release	Source
NSL-KDD	1,25,973	42	Continuous, Categorical, Binary	DoS, Probe, R2L, U2R	67,343	58,630	2009	Canadian Institute for Cybersecurity (CIC)
UNSW-NB15	25,40,044	51	Continuous, Categorical, Binary	Fuzzers, Analysis, Backdoors, DoS, Exploits, Generic, Reconnaissance, Shellcode, Worms	22,18,761	3,21,283	2015	Australian Centre for Cyber Security (ACCS)

The introduced intrusion detection system was tested based on important classification measures, such as the precision, Accuracy, Recall, F1-score, and Detection Rate (DR). Resembling the proportion of correctly labelled samples, Accuracy points at the fraction of the true positives amid the predicted positives, which is represented by Precision. The measures of the percentage of true positives correctly classified is Recall or True Positive Rate. The F1-score, the harmonic average of Precision and Recall balances the two to obtain an overall evaluation. Detection Rate concentrates on how well the system managed to identify instances of intrusion which is of paramount importance when it comes to minimizing neglected attacks. These criteria are based on the classifications conventionally used: True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN).

Table 2. Performance Comparison of NSL-KDD

Technique	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	Detection Rate (%)
Decision Tree (DT)	91.87	90.96	90.12	90.54	90.12
Random Forest (RF)	94.23	93.78	93.21	93.49	93.21
Support Vector Machine (SVM)	95.01	94.52	94	94.26	94
CNN–PSO–FMMNN (baseline hybrid)	96.35	95.89	95.42	95.65	95.42
Proposed CNN–QPSO–FMMNN	**98.12**	**97.85**	**97.51**	**97.68**	**97.51**

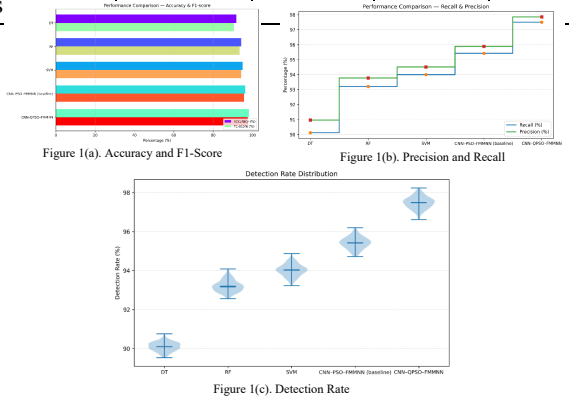

Figure 1(a). Accuracy and F1-Score

Figure 1(b). Precision and Recall

Figure 1(c). Detection Rate

Figure 1. Comparison of NSL-KDD

The performance records in Table 2 and Figure 1 show the good behavior of the proposed CNN-QPSO-FMMNN architecture in front of the traditional and hybrid baselines. DT classifier yielded an accuracy of 91.87 percent, and with the detection rate being 90.12 percent the type of classification is of a medium level and would not be useful in complex intrusion patterns. Other methods introduced to enhance performance are Random Forest (RF) and Support Vector Machine (SVM), which used such concepts as ensemble optimization and margin-based optimization, which attained accuracy rates of 94.23% and 95.01% respectively. The baseline hybrid model CNN–PSO–FMMNN further enhanced results to 96.35% accuracy, highlighting the benefit of deep feature extraction and metaheuristic-based feature optimization. The proposed CNN–QPSO–FMMNN attained the highest accuracy (98.12%), precision (97.85%), recall (97.51%), and F1-score (97.68%), indicating balanced improvements across detection capability and false positive minimization. This performance gain can be attributed to the QPSO's superior global search

capability, which improves optimal feature subset selection, and the FMMNN's ability to model non-linear decision boundaries effectively.

Table 3. Performance Comparison of UNSW-NB15

Technique	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	Detection Rate (%)
Decision Tree (DT)	89.34	88.72	87.95	88.33	87.95
Random Forest (RF)	92.46	92.01	91.67	91.84	91.67
Support Vector Machine (SVM)	93.78	93.25	92.9	93.07	92.9
CNN–PSO–FMMNN (baseline hybrid)	94.85	94.41	94.12	94.26	94.12
Proposed CNN–QPSO–FMMNN	**96.93**	**96.58**	**96.25**	**96.41**	**96.25**

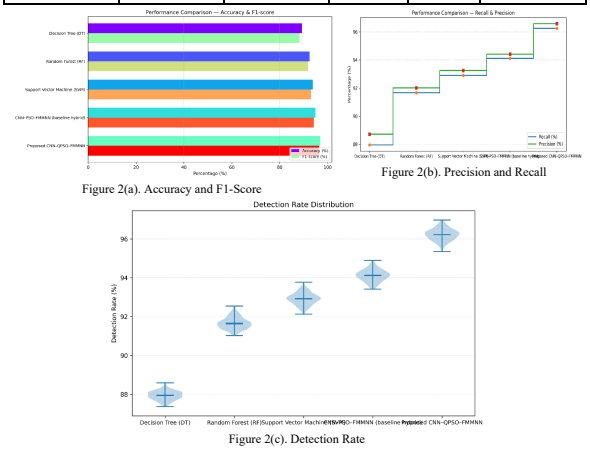

Figure 2(a). Accuracy and F1-Score

Figure 2(b). Precision and Recall

Figure 2(c). Detection Rate

Figure 2. Comparison of UNSW-NB15

Table 3 and Figure 2 reveal a similar trend on the more challenging UNSW-NB15 dataset, which contains modern and diverse attack vectors. DT achieved the lowest performance, with 89.34% accuracy and 87.95% detection rate, showing limited scalability for heterogeneous traffic. RF and SVM demonstrated incremental improvements with accuracies of 92.46% and 93.78%, respectively. The baseline CNN–PSO–FMMNN achieved 94.85% accuracy, demonstrating that deep-learning-driven feature extraction coupled with PSO-based selection enhances classification performance in contemporary intrusion scenarios. The presented CNN-QPSO-FMMNN performed significantly better at the accuracy percents of 96.93, with 96.25 detection rate, surpassing all baselines methods in all measures. The findings confirm the effectiveness of the proposed methodology when managing a wide variety of attack patterns and traffic profiles and, thus, increase the generalization capacity of NB-group intrusion detection systems that work with both legacy (NSL-KDD) and current (UNSW-NB15) datasets.

5. Conclusion and Future Research Work

The newly suggested CNN–QPSO–FMMNN system that incorporates CNN-based attribute extraction, Quantum-inspired Particle Swarm Optimisation (QPSO) to select the best features and Fuzzy Min-max Neural Network (FMMNN) to classify attributes has shown significant gains in intrusion detection performance of Wireless Sensor Networks and also cloud-based systems. Its robustness was confirmed with empirical tests on NSL-KDD and UNSW-NB15 datasets which reached up to 98.12 and 96.93 accuracy, respectively, and was able to save 28.6 percent training time compared to classical IDS. The high global search capability of the QPSO was also good to overcome the problem of early convergence, and FMMNN has the possibility of high adaptability to nonlinear decision boundaries and different types of patterns of attack due to the hyperbox-based learning. The framework is highly resistant to both known and zero-day attacks with only low false positives, so it is highly suitable in case of real-life resource constrained systems and in real time. This combination of deep feature learning, quantum-esque optimization and fuzzy neural classification provides a computationally cost-effective, scalable and highly accurate IDS solution, which pushes the state-of-the-art in the area of network security intelligence.

Future work will include implementation of CNN–QPSO–FMMNN in large scale distributed systems in real time, introducing blockchain based trust framework and expanding applicability of CNN–QPSO–FMMNN through continuous learning and multi-agent collaboration with IDS.

Reference

[1]. Mastelic, T., & Brandic, I. (2015). Recent trends in energy-efficient cloud computing. *IEEE Cloud Computing*, *2*(1), 40-47.

[2]. Moghaddam, F. F., Ahmadi, M., Sarvari, S., Eslami, M., & Golkar, A. (2015, May). Cloud computing challenges and opportunities: A survey. In *2015 1st International Conference on Telematics and Future Generation Networks (TAFGEN)* (pp. 34-38). IEEE.

[3]. Zhan, Z. H., Liu, X. F., Gong, Y. J., Zhang, J., Chung, H. S. H., & Li, Y. (2015). Cloud computing resource scheduling and a survey of its evolutionary approaches. *ACM Computing Surveys (CSUR)*, *47*(4), 1-33.

[4]. Rani, B. K., Rani, B. P., & Babu, A. V. (2015). Cloud computing and inter-clouds–types, topologies and research issues. *Procedia Computer Science*, *50*, 24-29.

[5]. Ramachandra, G., Iftikhar, M., & Khan, F. A. (2017). A comprehensive survey on security in cloud computing. *Procedia Computer Science*, *110*, 465-472.

[6]. Khan, N., & Al-Yasiri, A. (2016). Identifying cloud security threats to strengthen cloud computing adoption framework. *Procedia Computer Science*, *94*, 485-490.

[7]. Astri, L. Y. (2015). A study literature of critical success factors of cloud computing in organizations. *Procedia Computer Science*, *59*, 188-194.

[8]. Virupakshar, K. B., Asundi, M., Channal, K., Shettar, P., Patil, S., & Narayan, D. G. (2020). Distributed denial of service (DDoS) attacks detection system for OpenStack-based private cloud. *Procedia Computer Science*, *167*, 2297-2307.

[9]. Babu, B. M., & Bhanu, M. S. (2015). Prevention of insider attacks by integrating behavior analysis with risk based access control model to protect cloud. *Procedia Computer Science*, *54*, 157-166.

[10]. Yadav, R. M. (2019). Effective analysis of malware detection in cloud computing. *Computers & Security*, *83*, 14-21.

[11]. Rajendran, P. K., Muthukumar, B., & Nagarajan, G. (2015). Hybrid intrusion detection system for private cloud: a systematic approach. *Procedia Computer Science*, *48*, 325-329.

[12]. Idhammad, M., Afdel, K., & Belouch, M. (2018). Distributed intrusion detection system for cloud environments based on data mining techniques. *Procedia Computer Science*, *127*, 35-41.

[13]. Bamakan, S. M. H., Amiri, B., Mirzabagheri, M., & Shi, Y. (2015). A new intrusion detection approach using PSO based multiple criteria linear programming. *Procedia Computer Science*, *55*, 231-237.

[14]. Khan, N., & Al-Yasiri, A. (2016). Identifying cloud security threats to strengthen cloud computing adoption framework. *Procedia Computer Science*, *94*, 485-490.

[15]. Kavitha, D., & Thejas, S. (2024). Ai enabled threat detection: Leveraging artificial intelligence for advanced security and cyber threat mitigation. IEEE Access.

[16]. Idouglid, L., Tkatek, S., Elfayq, K., & Guezzaz, A. (2024). Next-gen security in IIoT: integrating intrusion detection systems with machine learning for industry 4.0 resilience. International Journal of Electrical & Computer Engineering (2088-8708), 14(3).

[17]. Rao, D. D., Waoo, A. A., Singh, M. P., Pareek, P. K., Kamal, S., & Pandit, S. V. (2024). Strategizing IoT network layer security through advanced intrusion detection systems and AI-driven threat analysis. Full Length Article, 12(2), 195-95.

[18]. Kim, H. R., & Song, H. M. (2024). Lightweight IDS Framework Using Word Embeddings for In-Vehicle Network Security. J. Wirel. Mob. Networks Ubiquitous Comput. Dependable Appl., 15(2), 1-13.

[19]. Farrukh, Y. A., Wali, S., Khan, I., & Bastian, N. D. (2024). Ais-nids: An intelligent and self-sustaining network intrusion detection system. Computers & Security, 144, 103982.

[20]. Ahmed, M. A. O., Abdelsatar, Y., Alotaibi, R., & Reyad, O. (2025). Enhancing Internet of Things security using performance gradient boosting for network intrusion detection systems. Alexandria Engineering Journal, 116, 472-482.

[21]. Hizal, S., Cavusoglu, U., & Akgun, D. (2024). A novel deep learning-based intrusion detection system for IoT DDoS security. Internet of Things, 28, 101336.