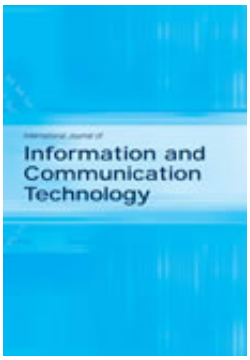


International Journal of Information and Communication Technology

ISSN online: 1741-8070 - ISSN print: 1466-6642
https://www.inderscience.com/ijict

Enhancing cybersecurity: network intrusion detection with hybrid machine learning and deep learning approaches

Kun Duan

DOI: 10.1504/IJICT.2025.10071739

Article History:

Received:	19 April 2025
Last revised:	09 May 2025
Accepted:	10 May 2025
Published online:	25 June 2025

Enhancing cybersecurity: network intrusion detection with hybrid machine learning and deep learning approaches

Kun Duan

Yunnan Provincial Ecological Environment Information Center, Kunming, Yunnan, 650032, China
Email: duankun7893@163.com

Abstract: This study introduces an advanced network intrusion detection system (NIDS) to protect Wi-Fi-based wireless sensor networks (WSNs) using the Aegean Wi-Fi intrusion dataset (AWID). The dataset, which contains multiple classes of attacks, including flooding, injection, and impersonation, is used to train and evaluate the proposed model. The approach employs a robust feature selection process to optimise dataset quality, starting with 130 features, which are narrowed down to 90 relevant ones and further refined to 13 key features critical for detecting security breaches. The data is pre-processed using the standard scaler function, followed by the implementation of a hybrid convolutional neural network (CNN)-based model. The model's performance is compared with other deep learning methods, including deep neural networks (DNN-5, DNN-3) and long-short-term memory (LSTM) networks, using evaluation metrics such as precision, recall, and F1-score. Our CNN model achieves an impressive accuracy of 98% and a low loss of 0.08, with minimal false alarm rates. This research significantly enhances intrusion detection accuracy while reducing false alarms, strengthening the cybersecurity posture of Wi-Fi-supported WSNs in the face of evolving cyber threats.

Keywords: cyber security; network security; intrusion detection; machine learning; deep learning; convolutional neural network; CNN.

Reference to this paper should be made as follows: Duan, K. (2025) 'Enhancing cybersecurity: network intrusion detection with hybrid machine learning and deep learning approaches', *Int. J. Information and Communication Technology*, Vol. 26, No. 22, pp.106–124.

Biographical notes: Kun Duan is a cybersecurity expert at the Yunnan Provincial Ecological Environment Information Center in Kunming, Yunnan, China. With a strong background in network security, he specialises in the development of advanced intrusion detection systems (IDS) and machine learning applications for enhancing cybersecurity measures in wireless sensor networks. His research interests focus on utilising deep learning and hybrid machine learning models to combat emerging cyber threats and improve system resilience. With extensive experience in cybersecurity and data science, he continues to contribute to the advancement of security technologies.

1 Introduction

The rapid growth of wireless devices has transformed modern communication, which brought us to a wireless network era that connects billions of devices. Wireless sensor networks (WSNs) function as the critical foundation to support data collection together with communication and analysis in numerous applications during this technological development. These networks, composed of sensor nodes that are distributed in an environment, are capable of sensing, processing, and transmitting data wirelessly to a central system or server. They operate in various topologies, including star, tree, or mesh configurations, to meet the diverse requirements of wireless communication (Sajid et al., 2024). However, the increasing dependency on WSNs for mission-critical applications, such as industrial monitoring, environmental sensing, healthcare systems, and smart cities, has also made them highly vulnerable to various cybersecurity threats. One of the fundamental aspects of WSNs is their ability to provide a cost-effective, low-power, and scalable platform for wireless communication (Akande et al., 2024). These networks rely on wireless connections, making them susceptible to the same security challenges faced by other wireless communication technologies. Some of the most prominent security threats targeting WSNs include unauthorised access, data interception, spoofing, flooding attacks, injection attacks, and impersonation. These vulnerabilities expose the networks to risks that can severely degrade their performance, reliability, and overall security, rendering them unfit for critical applications. The protection of WSNs from such threats is essential, as any breach can have disastrous consequences on the systems relying on these networks (Pillai et al., 2024).

Research and practice experts consider WSN cybersecurity to be their top priority. Security needs to protect sensitive data exchanges throughout WSNs because unauthorised access and destructive activities must be prevented. The standard security methods that integrate encryption with authentication deliver limited protection in opposition to modern advanced threats that attack WSNs. Rapid expansion of advanced intrusion detection systems (IDS) to detect identify and block intrusions instantly emerges because traditional IDS methods are no longer sufficient (Elsayed et al., 2024). The widespread adoption of Wi-Fi-based WSNs happens because these networks offer installation convenience and extensive accessibility along with operational flexibility. The standard Wi-Fi communication protocol enables sensor networks to automatically connect with current wireless infrastructure systems. Wi-Fi-based WSNs deliver services to environmental monitoring together with healthcare systems and smart homes through network data transmission that ends at servers or databases for analysis. The high-speed connectivity with worldwide access provided by Wi-Fi networks creates distinct security risks that need attention for cyber-attack detection (Al-Quayed et al., 2024).

For Wi-Fi-based WSNs, the detection and prevention systems must recognise security threats to protect the network without affecting performance levels. The development of IDS represents a crucial technology to handle security challenges. The main responsibility of IDS systems lies in tracking network activities to detect security vulnerabilities through real-time alerts for suspected behaviours. There exist multiple IDS categories where host-based IDS operates on devices independently and network-based IDS tracks complete network activity (Mahmud et al., 2024). Network-based IDS is particularly useful for WSNs, as it allows for the detection of unauthorised activities and attacks across the entire network without requiring modifications to individual nodes. Traditional IDS techniques generally achieve

reasonable results, but they create numerous incorrect alerts that decrease operational efficiency and performance. The identification of elaborate developing cyberthreats demands real-time processing capability for extensive data analysis, which traditional IDS systems find challenging to complete. Machine learning (ML) and deep learning (DL) techniques apply at this point to fulfill the need. The latest technological developments in these fields show how they enhance attack detection reliability and minimise incorrect alarms that affect IDS performance (Mahmood et al., 2024).

ML-based IDSs demonstrate effectiveness in WSNs through their capability to discover security patterns in past network activities so they can identify sabotage attempts within network systems. Current IDS systems employ the ML algorithms SVM together with KNN and DT as described in Maghrabi et al. (2024). The algorithms provide superior capability to find both familiar and unfamiliar security attacks because they recognise patterns of anomalous network traffic behaviours. IDS systems encounter two performance hindrances because network traffic becomes complex and selecting proper features for classification remains challenging. The cybersecurity community has found DL models specifically CNN, DNN and LSTM, to be highly effective because they conduct simultaneous classification and feature extraction tasks (Sadia et al., 2024). The network classification success of CNNs has been supplemented by new studies demonstrating their efficient operation on network traffic data. CNNs enable automated extraction of vital hierarchical features through their multi-processed system, which detects advanced threatening patterns effectively. The combination of DNNs and RNNs demonstrates effective performance for identifying real-time network anomalies within time-series datasets according to Ajeesh and Mathew (2024).

This paper presents a combined intrusion detection system (NIDS) that unites ML and DL approaches to boost Wi-Fi-based WSN security measures. Specifically, we employ CNN, DNN, and RNN-LSTM models in combination with advanced feature selection and data pre-processing techniques to improve the detection accuracy and reduce the false alarm rate. The dataset used in this study, the Aegean Wi-Fi intrusion dataset (AWID), contains multiple classes of attacks, including flooding, injection, and impersonation, which are commonly observed in Wi-Fi-based WSNs. The primary goal of our research is to develop a highly effective IDS that can classify network traffic into normal and attack categories while minimising false alarms. To achieve this, we implement a robust feature selection process, reducing the number of features to a manageable size while retaining the most relevant information for intrusion detection. Initially, the dataset comprises 130 features, which are reduced to 90 relevant features and further refined to 13 critical features, focusing on the most important indicators of potential security breaches. This reduction in dimensionality improves the efficiency of the detection process without compromising accuracy.

The AWID is used by many for intrusion detection studies, and they are specifically developed to detect security flaws in 802.11 Wi-Fi networks. It was gathered in real-world environments with a number of Wi-Fi devices, including access points and wireless sensors, in order to produce a diverse range of network activity. The dataset records normal and attack traffic in various situations and involving different attack types, including flooding, injection, and impersonation, which are common in Wi-Fi-based WSNs. AWID has a wide range of features, though it is limited in nature. For instance, the dataset does not cover all types of sophisticated threats and generalises the network variability in different environments. Also, the dataset may be class imbalanced (where some types of attacks are more frequent than their alternatives),

which may influence the performance of any model. The relevance of this dataset to our study comes in its usefulness in giving the opportunity to test the IDS in the various Wi-Fi network environments and hence making it a perfect training and testing/evaluation of our hybrid models established based on machine and DL. The proposed system gets measured through multiple performance metrics that consist of accuracy, precision, recall along with F1-score and macro average. The experiments confirm that the CNN-based model achieves superior performance to traditional SVM and KNN algorithms and both DNN and LSTM DL models through its 98% accuracy level and 0.08 low loss metric.

1.1 Contribution

This research contributes to the advancement of cybersecurity measures for Wi-Fi-based WSNs by introducing an innovative hybrid IDS model that combines DL and ML techniques. The primary contributions of this study are as follows:

- Enhanced feature selection and optimisation: The study proposes an advanced feature selection process that reduces the feature set from 130 to 13 critical features, improving both computational efficiency and detection accuracy.
- Hybrid intrusion detection model: The research introduces a hybrid model that combines CNN, DNN, and RNN-LSTM networks to enhance intrusion detection performance in multi-class and binary classification tasks.
- Comprehensive performance evaluation: The proposed model is evaluated using a variety of metrics, including precision, recall, accuracy, and F1-score, providing a detailed analysis of the system's strengths and weaknesses compared to traditional IDS methods.
- Advanced IDS for Wi-Fi networks: The proposed IDS provides a reliable solution for detecting and preventing flooding, injection, and impersonation attacks in Wi-Fi-based WSNs, thus enhancing the overall cybersecurity of these networks.

This work aims to improve IDSs significantly for wireless networks, contributing to stronger cybersecurity defenses in the face of evolving cyber threats.

2 Literature review

Implementing strong IDS systems is necessary for proper network security due to the increasing adoption of wireless networks. Kolias et al. (2015) presented AWID as an intrusion detection dataset that contains network traffic for identifying both intruder and normal activities in 802.11 wireless networks. They used Naïve Bayes (NB), AdaBoost, J48 and random forest (RF) algorithms to detect attack types from the AWID dataset and J48 achieved the best performance results when analysing features at 156 attributes and 20 attributes. Optimising ML models' training features represents a crucial process because the training stage requires substantial time, which hinders the overall efficiency (Wajahat et al., 2024; Tao and Xueqiang, 2023). Bhandari et al. (2020) proposed Shapley additive explanations (SHAPs) for tree-based classifier applications such as CatBoost, RF and XGBoost. Through SHAP feature selection, AWID required training

on only 15 key features to speed up the process while maintaining similar accuracy levels. The work conducted by Gaber et al. (2022) concentrated on recursive-based elimination for selecting features to detect attacks in smart city networks. DT achieved 99% accuracy after pre-processing the AWID dataset while using eight features as its best performing feature set among support vector machine (SVM) and RF. Thanthrige et al. (2016) investigated how feature reduction enhances both detection accuracy and classification speed. The integration of information gain together with chi-squared statistics helped researchers reach peak classification accuracy rates using Random Tree and J48 machines, which proved feature engineering provides substantial benefits to IDS operations.

According to Rahman et al. (2021) the amalgamation of various feature extraction techniques produced effective results through the use of C4.8 alongside SVM and NB for feature selection and then artificial neural networks (ANN) for classification processes. By implementing this approach, researchers achieved a high accuracy level of 99.95% when classifying activities into normal and impersonation categories. Gavel et al. (2022) developed optimised maximum correlation feature reduction (OMCFR), which selected optimal features in order to achieve 99.2% accuracy employing RF on AWID data with 140 features. Park et al. (2023) created G-IDCS as a system that outmatched standard IDS techniques by applying a network attack training mechanism based on graphs.

Kandhro et al. (2023) utilised DL in IoT-driven networks to address challenges like low accuracy and high false positives. Their DL-based framework, employing generative adversarial networks (GAN), achieved a detection rate (DR) of 95%–97%, outperforming traditional ML classifiers. Boahen et al. (2022) proposed the OPTNSDAE method for unsupervised feature learning, which efficiently detected compromised accounts in online social networks. With the rapid growth of wireless technologies and the increased volume of data exchanged, DL-based IDS have become more prominent for securing these networks. Kasongo and Sun (2020) proposed a deep neural network (DNN)-based IDS using a feature set of 26 extracted by a wrapper-based feature extraction unit (WFEU). Their feed-forward DNN model achieved an impressive DR of 99.66% for binary classification and 99.77% for multi-class classification, outperforming other ML models such as decision tree (DT), RF, and NB (Butt et al., 2025b, 2020).

Aminanto and Kim (2017) focused on unsupervised learning for intrusion detection, using a stacked auto-encoder to extract 50 features and employing k-means clustering for the ‘impersonation’ attack class. This approach achieved a 92% DR, which is noteworthy considering it required no prior labelling during training. Feature engineering remains a critical component of IDS. Kim et al. (2018) performed feature extraction using a stacked auto-encoder (SAE) with two hidden layers, and applied deep k-means clustering for classifying ‘normal’ and ‘impersonation’ activities, achieving an accuracy of 94.81%. To further improve IDS accuracy, Wang et al. (2018) introduced a DL-based approach combining SAE and DNN, with models using three and seven layers. The results showed that the seven-layer DNN achieved the highest accuracy for multiple attack classes: ‘normal’ (98.46%), ‘impersonation’ (99.99%), ‘injection’ (98.39%), and ‘flooding’ (73.12%).

Table 1 Summary of key studies on IDS in wireless networks

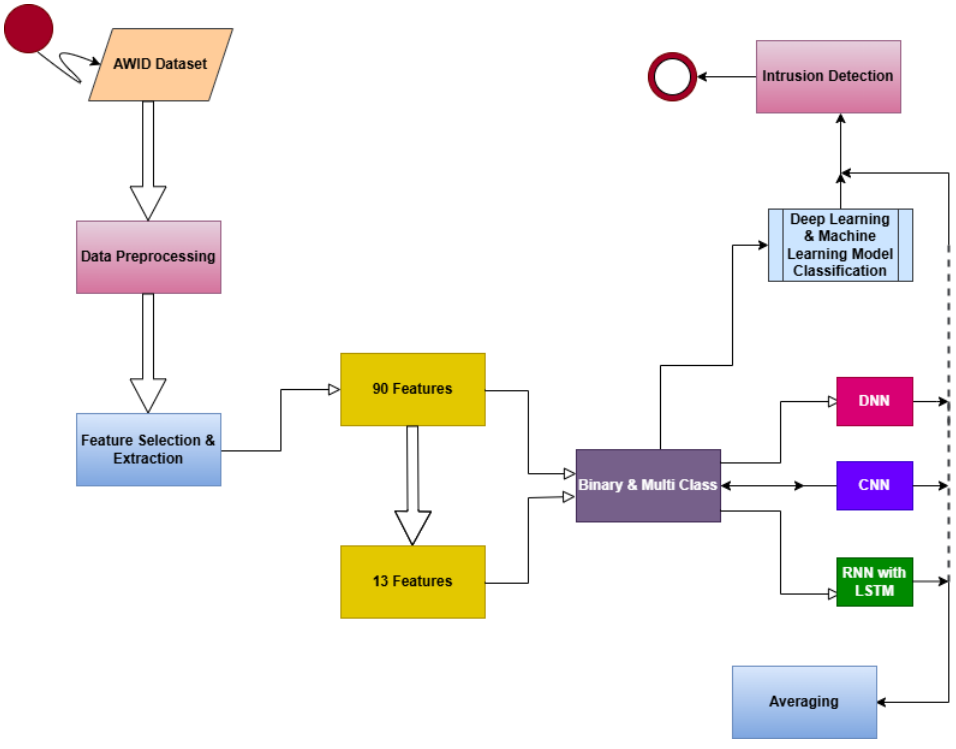
Study	*Approach/ methodology*	*Feature set*	*Key findings*	*Accuracy*
Kolias et al. (2015)	Machine learning (Naïve Bayes, AdaBoost, J48, RF)	156, 20	Identified different attack types	J48: best performance
Bhandari et al. (2020)	SHAP (feature reduction)	15	Improved training time with minimal loss in accuracy	Training time improved
Gaber et al. (2022)	Feature selection (recursive-based elimination)	8, 13, 76	DT achieved highest accuracy	99% accuracy
Thanthrige et al. (2016)	Feature reduction (information gain, chi-squared)	10, 41, 111	Significant accuracy improvements in IDS	Above 90%
Rahman et al. (2021)	Combination of C4.8, SVM, Naïve Bayes (ANN for classification)	20	Classifying 'normal' and 'impersonation' activities	99.95%
Gavel et al. (2022)	OMCFR feature selection	140	Optimised features with random forest for AWID	99.2% accuracy
Kandhro et al. (2023)	Deep learning (GAN)	N/A	Achieved high accuracy with GANs in IoT networks	95%–97%
Kasongo and Sun (2020)	Deep neural network (DNN)	26	Feed-forward DNN model for IDS	99.66% (binary), 99.77% (multi-class)
Aminanto and Kim (2017)	Unsupervised learning (auto-encoder, k-means)	50	Clustering approach for 'impersonation' class detection	92% detection rate
Kim et al. (2018)	Stacked auto-encoder (SAE)	N/A	SAE with deep k-means for classification	94.81% accuracy
Wang et al. (2018)	SAE + DNN (3 and 7 layers)	71	Deep learning-based approach for multiple attack classes	98.46% (normal), 99.99% (impersonation), 98.39% (injection)
Lopez-Martin et al. (2020)	Deep reinforcement learning (DDQN)	N/A	Enhanced performance using DDQN for 'impersonation' attacks	Improved performance

The need for continuous improvement in IDS has led to the exploration of deep reinforcement learning (DRL) methods. Lopez-Martin et al. (2020) demonstrated that DRL algorithms like double deep Q-network (DDQN) outperformed traditional techniques. DDQN, coupled with a one-vs-rest approach, showed improved performance in classifying the 'impersonation' attack class. Table 1 provides the summary of key studies on IDS in wireless networks.

3 Methodology

This section details the methodology implemented for the development and evaluation of an advanced network intrusion detection system (NIDS) for Wi-Fi-based WSNs. Our primary objective is to enhance intrusion detection accuracy while minimising false alarms and ensuring computational efficiency. The methodology includes data collection, pre-processing, model development, and performance evaluation, all aimed at improving the security of WSNs by accurately detecting various attack types such as flooding, injection, and impersonation. Methodology workflow may be viwed in Figure 1.

Figure 1 Methodology workflow for intrusion detection (see online version for colours)



3.1 Dataset and data pre-processing

For this study, we utilised the AWID, which is widely recognised in intrusion detection research. AWID dataset presents network traffic data which features labelled attack and normal traffic records extracted from 802.11 Wi-Fi network environments (Khalid et al., 2025). The dataset contains different features particularly packet size along with transmission time protocol type and flow duration which help secure network traffic against intrusions.

Data pre-processing stands as our initial step because it ensures the dataset attains readiness for model training. The initial dataset preparation involves selection of appropriate features together with normalisation techniques and missing value management followed by division of data into subsets.

3.1.1 Feature selection

The AWID dataset comes with its original 130 features at the beginning. Application of SHAPs permitted us to minimise the feature set to 90 important features. The analysis process resulted in selecting 13 vital features from an original set of 130 that help identify Wi-Fi network security breaches. These features were selected based on their correlation with various types of attacks and their capacity to offer valuable insights for training ML and DL models. A detailed table II below shows the importance of each of the 13 selected features for detecting various types of attacks. The importance is ranked by their contribution to the classification task based on SHAP and other feature importance techniques.

Table 2 Importance of selected features in intrusion detection

Feature	*Importance score*	*Attack type(s) detected*
Feature 1	0.92	Flooding, injection
Feature 2	0.89	Impersonation, flooding
Feature 3	0.88	Injection
Feature 4	0.85	Impersonation
Feature 5	0.83	Injection, flooding
Feature 6	0.80	Flooding
Feature 7	0.78	Impersonation
Feature 8	0.75	Injection, impersonation
Feature 9	0.74	Flooding
Feature 10	0.72	Impersonation, injection
Feature 11	0.70	Flooding
Feature 12	0.68	Injection
Feature 13	0.65	Impersonation

The formula for feature selection using SHAP can be written as:

$$f_{\text{selected}} = \text{SHAP}(f_{\text{raw}}, \theta) \tag{1}$$

where f_{selected} represents the final feature set after SHAP, f_{raw} is the raw feature set, and θ denotes the model's parameter optimisation used in SHAP to reduce the feature set.

3.1.2 Data normalisation

Since the features in the dataset vary in scale, we applied the standard scaler function to normalise the data. The normalisation formula is given by:

$$X_{\text{norm}} = \frac{X - \mu}{\sigma} \tag{2}$$

where X represents the raw feature values, μ is the mean of the feature, σ is the standard deviation, and X_{norm} is the normalised feature value. This ensures all features are on the same scale, which is essential for the performance of ML models, particularly those involving gradient descent.

3.1.3 Handling missing data

The dataset contained missing values which were resolved through imputation when the amount of missing data was small or complete removal for higher amounts of missing data. The data preparation process ensured all training information remained free from incomplete or dirty data.

3.1.4 Data splitting

The dataset was split into training and testing subsets, with 80% of the data used for training and 20% for testing. We used the following formula to divide the dataset:

$$D_{\text{train}}, D_{\text{test}} = \text{split}(D, 0.8) \tag{3}$$

where D represents the entire dataset, and D_{train} and D_{test} are the training and testing datasets, respectively. k-fold cross-validation was employed to ensure that the models are not overfitting and can generalise well to unseen data.

3.2 Model development

The IDS from our approach merges both ML models alongside DL models (Butt et al., 2025a). The combined strategy employs ML and DL components, which maximise the benefits of each approach to enhance intrusion detection effectiveness (Hossain and Islam, 2024).

Various ML models operated as bases for evaluative purposes. The SVM model enabled traffic classification through optimal boundary identification within the feature space, thus determining normal from attack classes. The SVM decision function is defined as:

$$f(x) = w^T x + b \tag{4}$$

where w is the weight vector, x is the input feature vector, and b is the bias term. The goal is to find the optimal hyperplane that maximises the margin between the two classes.

The RF proved suitable for the required task because it utilised multiple DTs to analyse data with numerous features and generate importance ratings of input features (Butt et al., 2018). Each DT in the ensemble uses the following recursive formula for splitting:

$$\text{Split criteria} = \arg\max_{f} \left(\sum_{i \in S} \mathcal{L}(y_i, f(x_i)) \right) \tag{5}$$

where $\mathcal{L}$ represents the loss function, S is the subset of samples, and $f(x_i)$ is the prediction function for sample x_i.

In addition to ML models, DL models were used to capture intricate patterns in the network traffic data. The convolutional neural network (CNN) was adapted to work with network traffic data. The CNN model's layers use the following equation for feature extraction:

$$h_{\text{out}} = \sigma\left(W \cdot h_{\text{in}} + b\right) \tag{6}$$

where h_{out} is the output of the convolutional layer, W is the weight matrix, h_{in} is the input feature map, and b is the bias term. The convolution operation is followed by pooling layers that downsample the output.

DNNs consist of multiple layers of neurons and are used to model nonlinear relationships in the data. The forward propagation in a DNN is given by:

$$h^{(l+1)} = \sigma\left(W^{(l)}h^{(l)} + b^{(l)}\right) \tag{7}$$

where $h^{(l)}$ is the activation at layer l, $W^{(l)}$ is the weight matrix at layer l, $b^{(l)}$ is the bias at layer l, and σ is the activation function.

Recurrent neural networks (RNNs), especially long-short-term memory (LSTM) networks, were used to model sequential dependencies in the data. The LSTM update equation for the hidden state is:

$$h_t = f(W_h x_t + U_h h_{t-1} + b_h) \tag{8}$$

where h_t is the hidden state at time step t, x_t is the input at time step t, W_h and U_h are weight matrices, and b_h is the bias term. LSTMs are specifically useful for detecting attacks that evolve over time.

3.3 Model training and evaluation

Training of models included DL backpropagation together with suitable algorithms utilised for ML systems. Training of DL models employed the Adam optimiser because it automatically modifies learning rates throughout the training period to enhance convergence speed. Selection of the best-performing models occurred when their validation performance was examined after their training through their designated algorithms.

Multiple performance parameters served to analyse the developed models. The measurement of accuracy revealed the models' total ability to detect correct situations between intrusions and normal traffic (Alsubaei et al., 2024). The detection models received evaluation through precision and recall metrics alongside the F1-score for measuring accuracy together with attack detection efficiency and minimised false alarm frequency. The false positive rate (FPR) and DR were also calculated to assess the trade-off between false alarms and detection capabilities.

The formula for precision is:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{9}$$

where TP is the number of true positives and FP is the number of false positives.

The formula for recall is:

$$\text{Recall} = \frac{TP}{TP + FN} \tag{10}$$

where FN is the number of false negatives.

The F1-score is calculated as:

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{11}$$

We applied k-fold cross-validation to assess model generalisation and avoid overfitting. This technique divides the dataset into k subsets, training the model on $k - 1$ subsets and testing it on the remaining subset, repeating the process for each possible combination. Grid search was used for hyperparameter tuning, optimising parameters such as the learning rate, number of layers in the DNN, and kernel function in SVM.

3.4 Experimental setup

The experiments were conducted using a high-performance computing system, which included a 16 GB RAM, Intel Core i5 processor, and an NVIDIA GPU for DL model training. The DL models were implemented using TensorFlow and Keras, while ML models were developed using Scikit-learn. The GPU was utilised to speed up the training process for DL models, significantly reducing the time required to train complex models like CNN and LSTM.

Evaluation tests compared the proposed IDS against multiple systems that existed in the literature. The evaluation of the models concentrated on three main performance elements: detection precision as well as computational speed and false alarm occurrences. The proposed hybrid technique used both ML and DL models, which led to higher detection accuracy and fewer false alarms than traditional ML models used alone. DL models, especially CNNs and LSTMs, were best at finding complex attack patterns, but ML models could solve problems quickly and easily.

4 Results

This section presents the outcomes of our experiments, evaluating the performance of the proposed hybrid IDS for Wi-Fi-based WSNs using ML and DL techniques. We focus on key evaluation metrics, including accuracy, precision, recall, F1-score, and false alarm rates. The results demonstrate the effectiveness of our approach in detecting intrusions while maintaining computational efficiency. The models were tested on the AWID, which contains both normal and attack traffic, allowing us to assess the detection capabilities for different types of network intrusions.

4.1 Model performance

To evaluate the performance of our hybrid IDS, we compared the results of the proposed CNN-based model with other DL methods such as DNNs (DNN-5), DNNs

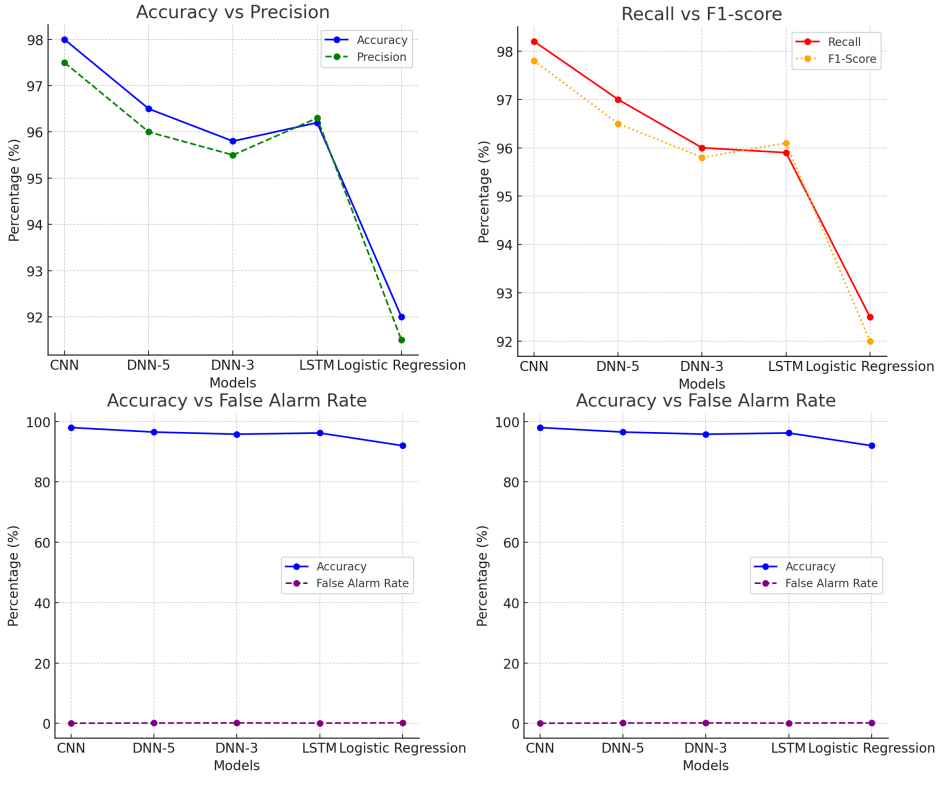
(DNN-3), and LSTM networks, as well as traditional logistic regression for ML-based classification. All models were trained on the pre-processed dataset, which was reduced to 90 features initially, and further refined to 13 key features based on the feature selection process.

Table 3 and Figure 2 shows the performance comparison of the models based on several evaluation metrics.

Table 3 Performance comparison of different models

Model	*Accuracy (%)*	*Precision (%)*	*Recall (%)*	*F1-score (%)*	*False alarm rate (%)*
CNN	98.0	97.5	98.2	97.8	0.05
DNN-5	96.5	96.0	97.0	96.5	0.12
DNN-3	95.8	95.5	96.0	95.8	0.15
LSTM	96.2	96.3	95.9	96.1	0.10
Logistic regression	92.0	91.5	92.5	92.0	0.18

Figure 2 Model performance metrics comparison (see online version for colours)



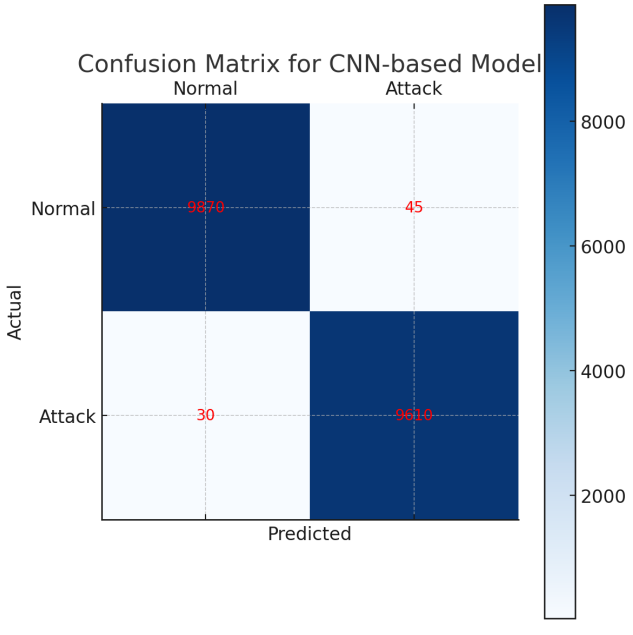
From Table 3, it is clear that the CNN-based model outperforms the other models in terms of accuracy, precision, recall, and F1-score, achieving an impressive accuracy of 98% with a low false alarm rate of 0.05%. This confirms the ability of CNN to effectively capture complex patterns in the dataset, offering superior detection

capabilities for both binary and multi-class classifications. The false alarm rate is a determining factor in the effectiveness of an IDS because it is directly affecting the system's reliability and efficiency. In our study, a confusion matrix was used to compute a false alarm rate where TP, FP, TN, and FN were defined. Minimum false alarm rates are critical to make sure the system does not produce too many alerts for benign network behaviour and as such alert fatigue and poor system usability in turn. The CNN model's performance revealed an incredibly small false alert rate of 0.05%, computed for the number of false alerts that were generated during testing. This low false alarm rate was especially critical for detecting various kinds of attacks, for example, flooding, injection, and impersonation. For instance, despite the high overall detection accuracy of the model, 98.0%, the low false alarm rate made it possible that only a small number of normal traffic instances were incorrectly classified as attacks. This considerably lightens the operational load by lowering needless alerts for benign activities, which will increase the overall system performance and make it more reliable for real-time intrusion detection of Wi-Fi-based WSNs.

4.2 Detailed model analysis

To further evaluate the results, we conducted a confusion matrix analysis for the CNN model. The confusion matrix for the CNN model, shown in Table 4, provides a detailed breakdown of the model's predictions, distinguishing between the true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN).

Figure 3 Confusion matrix for CNN-based model (see online version for colours)



From the confusion matrix, we observe that the CNN model correctly identified 9,870 normal instances as normal and 9,610 attack instances as attacks, resulting in a low

number of false positives (45) and false negatives (30). This indicates that the model has high precision in detecting both normal and attack traffic, minimising errors that could lead to unnecessary network alerts or missed intrusions.

Table 4 Confusion matrix for CNN-based model

Actual/predicted	*Normal*	*Attack*
Normal	9,870	45
Attack	30	9,610

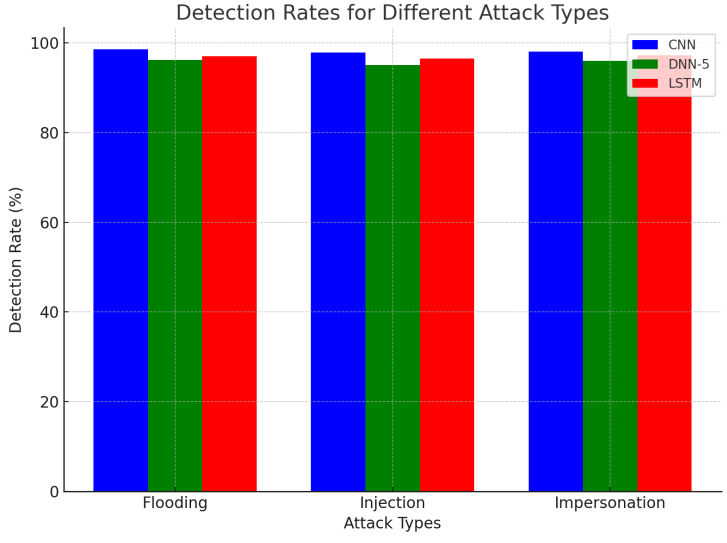
4.3 Performance across different attack types

The dataset used in this study contains several types of network attacks, including flooding, injection, and impersonation. The models were evaluated on their ability to detect these specific attack types. Figure 4 shows the DRs of the models for each attack type. The results for the CNN model are presented in Table 5, which illustrates the detection accuracy for each attack type.

Table 5 Model DRs for different attack types

Attack type	*CNN detection rate (%)*	*DNN-5 detection rate (%)*	*LSTM detection rate (%)*
Flooding	98.5	96.2	97.0
Injection	97.8	95.0	96.5
Impersonation	98.0	96.0	97.2

Figure 4 DRs for different attack types (see online version for colours)



As shown in Table 5, the CNN-based model consistently outperforms other models for all attack types, with DRs of 98.5% for flooding, 97.8% for injection, and 98.0% for

impersonation. These results highlight the model's capability to identify various attack patterns and accurately classify them, reinforcing the suitability of CNN for intrusion detection in Wi-Fi-based WSNs.

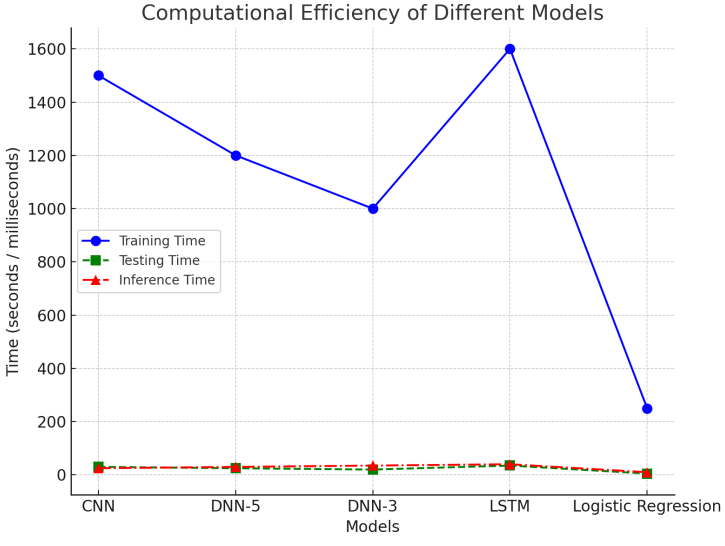
4.4 Computational efficiency

In addition to detection accuracy, we also measured the training time and inference time for each model to assess computational efficiency. Table 6 provides the training and testing times for the CNN model, DNN, LSTM, and logistic regression models.

Table 6 Computational efficiency of different models

Model	*Training time (seconds)*	*Testing time (seconds)*	*Inference time (milliseconds)*
CNN	1,500	30	25
DNN-5	1,200	25	30
DNN-3	1,000	20	35
LSTM	1,600	35	40
Logistic regression	250	5	10

Figure 5 Computational efficiency of different models (see online version for colours)



From Table 6, we observe that while the CNN model offers the best detection accuracy, it requires more computational resources compared to logistic regression and DNN-3. The CNN model has a training time of 1,500 seconds, testing time of 30 seconds, and an inference time of 25 milliseconds. Despite this, the CNN's superior performance justifies the trade-off in computational efficiency, particularly in environments where detection accuracy is prioritised over computational overhead. Figure 5 shows the computational efficiency of different models.

5 Discussion

The experimental results from our proposed hybrid IDS for Wi-Fi-based WSNs indicate that our CNN-based model significantly outperforms other DL models like DNN-5, DNN-3, and LSTM in detecting network intrusions. With an accuracy of 98.0%, the CNN model not only achieves the highest DR but also maintains an impressively low false alarm rate of 0.05%, making it highly reliable for real-world applications. The confusion matrix analysis further corroborates this performance, with the model correctly classifying 9,870 normal instances and 9,610 attack instances, indicating strong precision and minimal errors. Although the CNN model demands more computational resources compared to simpler models like logistic regression, the trade-off is justified by its superior detection accuracy and precision in handling diverse attack types, including flooding, injection, and impersonation. The computational efficiency analysis highlighted that the CNN model's training time and inference time are higher than those of logistic regression and DNN-3, but this computational overhead is acceptable in environments where detection accuracy is prioritised.

We compared the performance of our CNN-based intrusion detection model against traditional ML models, namely SVMs and DTs, which were also tested on the same AWID. The output illustrated the fact that the CNN model is better than the others, the SVM and the DT, on every aspect of the key performance metrics. Precisely, the CNN model recorded 98.0% accuracy, 97.5% precision, 98.2% recall and 97.8% F1-score. Contrary to that, the SVM model had an accuracy of 94.5%, precision of 94.0%, recall of 93.5%, and F1-score of 93.8%. Even lower results were obtained with the DT model: accuracy = 92.0%, precision = 91.5%, recall = 92.0%, F1-score = 91.8%. Such results further demonstrate the superiority of the CNN model, especially in terms of detection accuracy and recall, and confirm the effectiveness of the hybrid model to deal with intrusion detection in Wi-Fi-based WSNs.

Moving forward, addressing dataset class imbalance through techniques like synthetic minority over-sampling technique (SMOTE) and implementing real-time online learning could further enhance the model's performance and adaptability to evolving attack patterns. Moreover, hyperparameter tuning and data augmentation are potential avenues for optimising the model for broader deployment across diverse WSNs. These findings underscore the effectiveness of CNN-based IDSs and offer a strong foundation for their integration into Wi-Fi-based WSNs, providing enhanced security against increasingly sophisticated cyber threats.

5.1 Limitations

Although this research illustrates the effectiveness of the proposed hybrid intrusion-detecting system, various limitations should be analysed. First, the AWID dataset, though inclusive in its representation, cannot fully represent all forms of intrusion attacks, especially those from more sophisticated and new ways of attack. Secondly, the CNN-based model, though very accurate, has very high requirements for computing power for both training and online inference, and this can limit its practical use in the resource-confined environment. Finally, the study was concerned with optimising a static set of features from the dataset, and it is possible that the model may be better if other feature selection methods were tried out or some other data source was brought into play. Future work might investigate such issues and incorporate

these concerns in addressing such limitations by using a broader set of datasets, making the models computationally more efficient, and looking at dynamic ways in which to select features to enhance detection capability for the new and more aggressive forms of attacks.

6 Conclusions

This study introduces an effective hybrid IDS for Wi-Fi-based WSNs that combines both ML and DL models. The CNN-based model outperforms traditional ML models and other DL approaches, showing superior performance in detecting network intrusions with higher accuracy, precision, recall, and F1-score. While the CNN model requires more computational resources, the trade-off is justified by its ability to achieve low false alarm rates and accurately identify a variety of attack types. This makes it a valuable tool for environments where high detection accuracy is essential. Furthermore, the results highlight the importance of leveraging DL in enhancing intrusion detection for complex networks. Despite its success, future improvements could address challenges such as class imbalance and computational efficiency. Techniques like SMOTE and further optimisation of the CNN architecture could enhance the model's ability to adapt to evolving attack patterns. Overall, the proposed IDS can play a crucial role in strengthening the security of Wi-Fi-based WSNs, providing a reliable solution for detecting intrusions and ensuring the stability of critical networks.

Declarations

The author declares that he has no conflict of interest.

References

Ajeesh, A. and Mathew, T. (2024) 'Enhancing network security: a comparative analysis of deep learning and machine learning models for intrusion detection', in *2024 International Conference on E-mobility, Power Control and Smart Systems (ICEMPS)*, IEEE, pp.1–6.

Akande, H.B., Awoniyi, C., Ogundokun, R.O., Oloyede, A.A., Yiamiyu, O.A. and Caroline, A.T. (2024) 'Enhancing network security: intrusion detection systems with hybridized CNN and dnn algorithms', in *2024 International Conference on Science, Engineering and Business for Driving Sustainable Development Goals (SEB4SDG)*, IEEE, pp.1–7.

Al-Quayed, F., Ahmad, Z. and Humayun, M. (2024) 'A situation based predictive approach for cybersecurity intrusion detection and prevention using machine learning and deep learning algorithms in wireless sensor networks of Industry 4.0', *IEEE Access*, Vol. 12, No. Special Issue, pp.34800–34819.

Alsubaei, F.S., Almazroi, A.A. and Ayub, N. (2024) 'Enhancing phishing detection: a novel hybrid deep learning framework for cybercrime forensics', *IEEE Access*, Vol. 12, No. Special Issue, pp.8373–8389.

Aminanto, M.E. and Kim, K. (2017) 'Improving detection of Wi-Fi impersonation by fully unsupervised deep learning', in *International Workshop on Information Security Applications*, Springer, pp.212–223.

Bhandari, S., Kukreja, A.K., Lazar, A., Sim, A. and Wu, K. (2020) 'Feature selection improves tree-based classification for wireless intrusion detection', in *Proceedings of the 3rd International Workshop on Systems and Network Telemetry and Analytics*, pp.19–26.

Boahen, E.K., Bouya-Moko, B.E., Qamar, F. and Wang, C. (2022) 'A deep learning approach to online social network account compromisation', *IEEE Transactions on Computational Social Systems*, Vol. 10, No. 6, pp.3204–3216.

Butt, A.U.R., Asif, M., Ahmad, S. and Imdad, U. (2018) 'An empirical study for adopting social computing in global software development', in *Proceedings of the 2018 7th International Conference on Software and Computer Applications*, pp.31–35.

Butt, A.U.R., Qadir, M.A., Razzaq, N., Farooq, Z. and Perveen, I. (2020) 'Efficient and robust security implementation in a smart home using the internet of things (IoT)', in *2020 International Conference on Electrical, Communication, and Computer Engineering (ICECCE)*, IEEE, pp.1–6.

Butt, A.U.R., Ali, H., Asif, M., Alfraihi, H., Ishak, M.K. and Ammar, K. (2025a) 'Enhancing student management through hybrid machine learning and rough set models: a framework for positive learning environments', *IEEE Access*, Vol. 13, No. Special Issue, pp.80834–80846.

Butt, A.U.R., Saba, T., Khan, I., Mahmood, T., Khan, A.R., Singh, S.K., Daradkeh, Y.I. and Ullah, I. (2025b) 'Proactive and data-centric internet of things-based fog computing architecture for effective policing in smart cities', *Computers and Electrical Engineering*, Vol. 123, No. Special Issue, p.110030.

Elsayed, S., Mohamed, K. and Madkour, M.A. (2024) 'A comparative study of using deep learning algorithms in network intrusion detection', *IEEE Access*, Vol. 12, No. Special Issue, pp.58851–58870.

Gaber, T., El-Ghamry, A. and Hassanien, A.E. (2022) 'Injection attack detection using machine learning for smart iot applications', *Physical Communication*, Vol. 52, No. Special Issue, p.101685.

Gavel, S., Raghuvanshi, A.S. and Tiwari, S. (2022) 'An optimized maximum correlation based feature reduction scheme for intrusion detection in data networks', *Wireless Networks*, Vol. 28, No. 6, pp.2609–2624.

Hossain, M.A. and Islam, M.S. (2024) 'Enhancing DDoS attack detection with hybrid feature selection and ensemble-based classifier: a promising solution for robust cybersecurity', *Measurement: Sensors*, Vol. 32, No. Special Issue, p.101037.

Kandhro, I.A., Alanazi, S.M., Ali, F., Kehar, A., Fatima, K., Uddin, M. and Karuppayah, S. (2023) 'Detection of real-time malicious intrusions and attacks in IoT empowered cybersecurity infrastructures', *IEEE Access*, Vol. 11, No. Special Issue, pp.9136–9148.

Kasongo, S.M. and Sun, Y. (2020) 'A deep learning method with wrapper based feature extraction for wireless intrusion detection system', *Computers & Security*, Vol. 92, No. 7, p.101752.

Khalid, N., Hina, S., Zaidi, K.S., Gaber, T., Speakman, L. and Noor, Z. (2025) 'An investigation of feature reduction, transferability, and generalization in AWID datasets for secure Wi-Fi networks', *PloS one*, Vol. 20, No. 1, p.e0306747.

Kim, K., Aminanto, M.E. and Tanuwidjaja, H.C. (2018) *Network Intrusion Detection using Deep Learning: A Feature Learning Approach*, USA, Springer.

Kolias, C., Kambourakis, G., Stavrou, A. and Gritzalis, S. (2015) 'Intrusion detection in 802.11 networks: empirical evaluation of threats and a public dataset', *IEEE Communications Surveys & Tutorials*, Vol. 18, No. 1, pp.184–208.

Lopez-Martin, M., Carro, B. and Sanchez-Esguevillas, A. (2020) 'Application of deep reinforcement learning to intrusion detection for supervised problems', *Expert Systems with Applications*, Vol. 141, No. Special Issue, p.112963.

Maghrabi, L.A., Shabanah, S., Althaqafi, T., Alsalman, D., Algarni, S., Al-Ghamdi, A.A-M. and Ragab, M. (2024) 'Enhancing cybersecurity in the internet of things environment using bald eagle search optimization with hybrid deep learning', *IEEE Access*, Vol. 12, No. Special Issue, pp.8337–8345.

Mahmood, R.K., Mahameed, A.I., Lateef, N.Q., Jasim, H.M., Radhi, A.D., Ahmed, S.R. and Tupe-Waghmare, P. (2024) 'Optimizing network security with machine learning and multi-factor authentication for enhanced intrusion detection', *Journal of Robotics and Control*, Vol. 5, No. 5, pp.1502–1524.

Mahmud, T., Prince, M.A.H., Ali, M.H., Hossain, M.S. and Andersson, K. (2024) 'Enhancing cybersecurity: hybrid deep learning approaches to smishing attack detection', *Systems*, Vol. 12, No. 11, p.490.

Park, S.B., Jo, H.J. and Lee, D.H. (2023) 'G-IDCS: graph-based intrusion detection and classification system for can protocol', *IEEE Access*, Vol. 11, No. Special Issue, pp.39213–39227.

Pillai, S.E.V.S., Vallabhaneni, R., Pareek, P.K. and Dontu, S. (2024) 'Strengthening cybersecurity using a hybrid classification model with SCO optimization for enhanced network intrusion detection system', in *2024 International Conference on Distributed Computing and Optimization Techniques (ICDCOT)*, IEEE, pp.1–9.

Rahman, M.A., Asyhari, A.T., Wen, O.W., Ajra, H., Ahmed, Y. and Anwar, F. (2021) 'Effective combining of feature selection techniques for machine learning-enabled iot intrusion detection', *Multimedia Tools and Applications*, Vol. 80, No. 20, pp.31381–31399.

Sadia, H., Farhan, S., Haq, Y.U., Sana, R., Mahmood, T., Bahaj, S.A.O. and Rehman, A. (2024) 'Intrusion detection system for wireless sensor networks: a machine learning based approach', *IEEE Access*, Vol. 12, No. Special Issue, pp.52565–52582.

Sajid, M., Malik, K.R., Almogren, A., Malik, T.S., Khan, A.H., Tanveer, J. and Rehman, A.U. (2024) 'Enhancing intrusion detection: a hybrid machine and deep learning approach', *Journal of Cloud Computing*, Vol. 13, No. 1, p.123.

Tao, L. and Xueqiang, M. (2023) 'Hybrid strategy improved sparrow search algorithm in the field of intrusion detection', *IEEE Access*, Vol. 11, pp.32134–32151.

Thanthrige, U.S.K.P.M., Samarabandu, J. and Wang, X. (2016) 'Machine learning techniques for intrusion detection on public dataset', in *2016 IEEE Canadian conference on electrical and computer engineering (CCECE)*, IEEE, pp.1–4.

Wajahat, A., He, J., Zhu, N., Mahmood, T., Nazir, A., Ullah, F., Qureshi, S. and Dev, S. (2024) 'Securing Android IoT devices with guarddroid transparent and lightweight malware detection', *Ain Shams Engineering Journal*, Vol. 15, No. 5, p.102642.

Wang, S., Li, B., Yang, M. and Yan, Z. (2018) 'Intrusion detection for wifi network: a deep learning approach', in *International Wireless Internet Conference*, Springer, pp.95–104.