Grenze International Journal of Engineering and Technology, June Issue

Machine Learning-based Intrusion Detection System for Public Wi-Fi

Renuka Kondabala[1], Karthik Vishnusree Chetlapalli[2], Konduparthi Ram Suhas[3], Patha Rahul[4] and Pulluri Koushik[5]
[1]Assistant Professor, Department of Information Technology Vallurupalli Nageswara Rao Vignana Jyothi Institute of Engineering and Technology, Hyderabad, Telangana, India,
Email: renukak@vnrvjiet,.in
[2-5]Department of Information Hyderabad, TG, India, Vallurupalli Nageswara Rao Vignana Jyothi Institute of Engineering and Technology
Email: karthikv.chetla@gmail.com, krsuhas7@gmail.com, rahulpatha2803@gmail.com, koushikpulluri321@gmail.com

***Abstract*— There is an increase in the use of public Wi-Fi in many places which is used for the convenient internet access. When cyberattacks occur, the networks get exposed to the attacks, like spoofing, DoS (Denial of Service) and man-in-the-middle, which harms the privacy and stability of the network. The conventional methods have some disadvantages. Intrusion Detection System (IDS) helps to enhance security of the networks. It inspects all the incoming network traffic by detecting its packet sizes, the connection duration and frequency. By applying supervised machine learning models, the detection rate can be further enhanced.**

***Keywords*— Intrusion Detection System, Public Wi-Fi, Machine learning, PCA, Cyber Security.**

I. PROBLEM STATEMENT

The widespread adoption of public Wi-Fi networks in airports, educational institutions, shopping malls, hotels, and cafes has significantly improved Internet accessibility and user convenience. The shared and open network are highly vulnerable to various kinds of cyber threats, like as Denial-of-Service (DoS) attacks, spoofing, rogue access points, man-in-the-middle (MITM) attacks, and unauthorized access attempts [3], [4], [7], [8]. Such kind of attacks can compromise user privacy, and disrupt the network services, and result in various substantial security risks.

By 2030, the wireless networks will support many devices creating a global digital ecosystem and continuous wireless connectivity where the use of IoT (Internet of Things) are increasingly equipped allowing device-to-device (D2D) communication [1][2]. Effective security policies are needed with both hardware and software solutions. The use of smart cards and VPNs come under hardware solutions whereas regular software updates, encryption techniques and IDS come under software solutions [3].

The system is crucial to detect and prevent cyberattacks early. Conventional security measures such as firewalls, authentication and encryption are widespread. Nevertheless, they usually have a hard time detection capability and efficiency, particularly energy consumption and bandwidth. The use of machine learning to form intelligent IDS solutions has generated so much buzz to fight these challenges. Machine learning helps the system study network traffic and find unusual patterns and identify abnormalities, enhancing their detecting ability when dealing with intrusion.

Grenze ID: 01.GIJET.12.2.348_26

Consequently, ML-based IDS now plays a crucial role in the current cybersecurity systems especially in 5G and wireless network infrastructures [4].
Conventional techniques like authentication tend to be incapable of balancing the efficiency of detection ability owing to its bandwidth and energy consumption.
IDS using ML is the only way to spot irregularities enhancing detection and used in modern cybersecurity frameworks [4]. IDS can spot unusual traffic patterns by training on labelled datasets. This learning creates a new adaptation to identify new and unseen attack types [5]. The use of explainable artificial intelligence (XAI) can enhance interpretability and accuracy during recent research [6]. Furthermore, data-driven techniques have identified "evil twin" attacks [7][8].
There are several challenges that exist despite advancements which is to manage high-dimensional data, which is reduced by applying dimensionality reduction to remove redundancy and retaining necessary features [9]. There is also the challenge of data imbalance due to oversampling and undersampling. It is essentially needed to constantly monitor the network traffic using anomaly detection alongside ML models to identify any deviations with some strategies like SNMP-based traffic analysis [10].
Recently, the accuracy and the reduction in false-alarm rates has improved by applying advanced models for extracting complex patterns in a high-dimensional network data yet, there is an issue of high computational resources which makes it impractical in real world [11][15]. There is also challenges of scaling and adaptability which have not been addressed [12][16]. The prevalence of datasets like UNSW-NB15 has some data imbalance and has a lack of diversity in the real-world environments causing influence on generalization and performance [13][14]. The performance was better when using both methods like feature selection to make it less complex [17]. Similarly, IoT ecosystem have also been used to support scaling but had extra overheads [18].
The use of deep learning models often struggle with high false-positive rates [19]. Hence, the use of hybrid machine learning models was suggested which consists of two or more algorithms but have to be fine-tuned for attaining better performance [20].
Modern security mechanisms like firewalls, Encryption protocols, Authentication systems, and virtual private networks (VPNs) provide an additional foundational layer of protection but are often inadequate for detecting sophisticated and evolving attacks in real time [11], [12], [16]. Signature-based intrusion detection systems are effective against previously known attack patterns; however, they lack the adaptability required to identify zero-day attacks and emerging threats in dynamic wireless environments [11], [15].
Hence, it is necessary to develop an efficient and scalable intrusion detection system designed for public Wi-Fi environments. This study provides use of multiple machine learning algorithms for detecting intrusions in public Wi-Fi network and thus provides the most suitable model for acquiring high accuracy and maintaining very low false-positive rates.

II. Research Gap

Even though various deep learning and machine learning approaches have been proposed for detection of intrusion, several challenges remain as it is.Existing solutions have primarily focuses on enterprise networks, IoT and cloud infrastructures, whereas fewer investigations only have addressed detection of intrusion specifically within the public Wi-Fi networks [3], [4], [18].
Deep learning-based approaches have provided excellent performance in intrusion detection but they require large computational resources and very large scale datasets for demonstration, this has become the limitation for this model [11], [15], [19]. Furthermore, many existing intrusion detection systems suffer from high false-positive rates, which may negatively affect network administration and security monitoring processes [13], [19].
One more limitation is the lack of comprehensive comparative studies involving multiple machine learning algorithms under an unified framework. While individual techniques such as Support Vector Machines, K-Nearest Neighbors, Decision Trees, and Neural Networks have been widely investigated, their relative effectiveness for public Wi-Fi intrusion detection remains insufficiently explored [4], [12], [16].
A research gap that exists in developing a machine learning-based intrusion detection framework that combines high detection accuracy, low computational overhead, reduced false-positive rates, and practical applicability for public Wi-Fi security.

III. Literature Review and Related Work

The use of the public Wi-Fi networks has become part of our life since a fast and simple internet like a cafe, airports, campuses, and shopping malls has become common. These networks are however open and dynamic,

thus being prone to cyberattacks. These open Wi-Fi networks allow attackers to employ numerous malicious actions. Such attacks have the potential of seriously impacting users' security and general stability of the network. Conventional security will no longer work well particularly when one is dealing with huge and dynamically operating wireless environments.

The study introduces an IDS using machine learning to create a more secure networks. The system traces network traffic, and examines the important variables, such as connection length, packet size. With supervised learning models we are able to differentiate between normal and malicious activity.

The IDS has ability to learn from data and improve over time. This means that not only can the known attack patterns be detected but also new and never seen threats. The system delivers the results that can be considered reliable and effective since they are measured in terms of accuracy, precision, etc. The system assists in reducing the risk of unauthorized access and network misuse because the threats in real-time are detected.

Conclusively, this study provides a sensible and effective way of how to protect open internet areas of Wi-Fi. The proposed system will increase the confidence of the users, enhance network security, and create a safer environment of wireless communication by using machine learning and real-time traffic analysis to recognize a broader scope of network violations.

TABLE I: LITERATURE TABLE

Author	Year	Method	Dataset	Accuracy	Limitation
Reyes et al.	2023	Two-stage ML IDS	AWID	96.4%	High complexity
Salah et al.	2024	ML for 5G IDS	AWID3	97.8%	Dataset-specific
Banakh et al.	2024	KNN-based Evil Twin Detection	IEEE 802.11	95.2%	Limited attack types
Proposed Work	2026	DT, SVM, KNN, NB, LR	NSL-KDD	99.88%	Future work required
Reyes et al.	2023	Two-stage ML IDS	AWID	96.4%	High complexity

IV. PROPOSED METHOD

This method presents a system using machine learning that is designed to monitor, analyze, and safeguard networks as they happen. This system is constructed on various modules which collaborate to intercept network traffic, know user behavior and detect potential cyber threats.

A. User Connection to Public Wi-Fi

Upon the user making a decision to have Wi-Fi connection to an access point, which is a public one, the process begins. Public Wi-Fi Networks are typically open and shared and several users can use the network simultaneously. Since it is open, these networks are susceptible to numerous cyber threats. There is all the user interaction with the network, leading to traffic, which can be analyzed on security.

B. Network Traffic Capture

This consists of source and destination addresses, the type of data being transferred, packet sizes and length of a connection and frequency of transfer of data. Notably, the system only captures metadata of the traffic, and not the content of the data. This assists in securing the privacy of users and also enables encrypted communication among users without transferring sensitive data.

C. Feature Extraction from Traffic Metadata

The traffic information obtained is sent to the feature extractor module and features based on traffic behavior are obtained. These characteristics are the behavioral properties of network usage and involve connection count, packet rate, average packet size, protocol distribution, session timing distributions. The process of feature extraction converts unstructured traffic into a structured format, which brings out patterns that may be used to determine intrusion.

D. Data Preprocessing

Here, the step contains data cleaning to eliminate noisy or incomplete data, categorical attributes encoding to numerical values which ensures that features are scaled in the same manner.

E. Machine Learning-Based Traffic Analysis

The cleaned data is then sent to a machine learning engine that uses trained supervised learning models. These models compare the current data patterns with known normal and malicious behaviors. Based on this, the system decides if the traffic is normal or if an intrusion is happening.

F. Classification Output

The classification module has two outputs as follows:

- Normal Traffic: This type of network activity is considered as normal and left without any interference.
- Intrusion Detected: Traffic is considered malicious, and it possible that there may be attacks like denial-of-service, spoofing, or unauthorized access.

G. Monitoring Dashboard and Reporting

The entire system will be completed with a monitoring dashboard, which displays the real-time network security status. Key metrics such as detection rates, number of the attacks detected and traffic behavior trends are shown on the dashboard. It applies the use of graphs, charts and reports to enable network administrators to be aware of the security situation, identify trends and make improved security decisions.

V. SYSTEM ARCHITECTURE

The following describes the recommended architecture where the dataset is loaded and prepared, then some important columns are retrieved and classify them using different algorithms. The results, along with metrics and visual displays, are shown through a user interface.

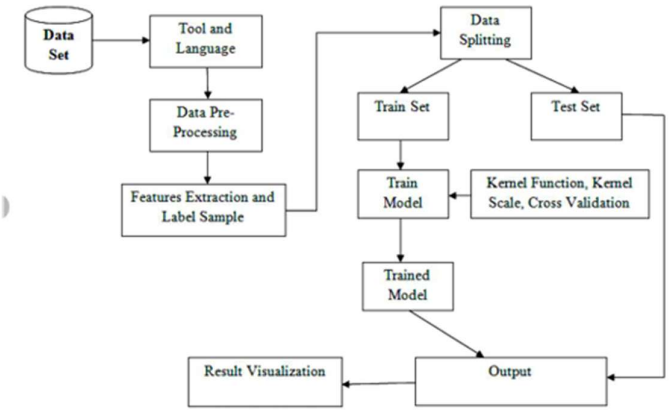

Fig 1: System Architecture

The diagram below shows the workflow process of a system for public Wi-Fi networks. It starts by collecting network traffic from the access points. This traffic is then prepared like cleaning the data and making it in a usable format, and scaling to a standard range. After that, important features are identified and is selected to make it more productive without any complication. The data is divided into 2 segments namely, the training and the testing with supervised learning models. The last model plays the role of the classifier; it examines the information flowing through a network to ascertain whether it is normal or an assault thereby assisting it in identifying possible cyber threats.

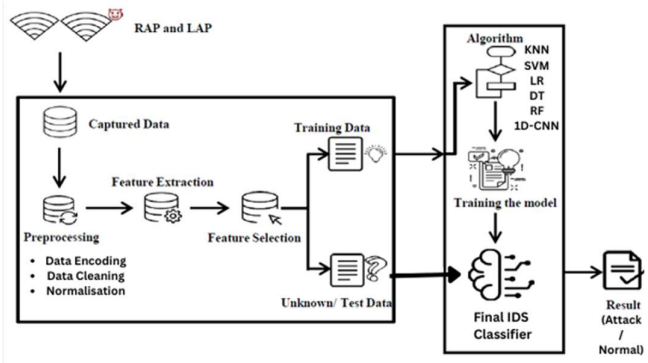

Fig: 2 Workflow Process

VI. Implementation Framework

In this project, this work presents a user-friendly and flexible IDS, which enhances the security of open internet networks. Cyber threats pose more threat to open public wireless environments due to their open nature and the manner in which the users are connected to other networks. This is why the development of an intelligent system making decisions and detecting the suspicious activity in real-time is in high demand.

The proposed system enables systematic comparison of the various algorithms that yields the most efficient mode of detecting network intrusions. Attacks can be detected by other methods by matching the attack with known attack patterns and this restricts their capability to identify novel threat. Instead, our method will label attacks based on traffic on networks. This will enable the system to identify normal and malicious actions and adjust itself to new kinds of threats automatically.

The system assists in the different classification models like:

- Support Vector Machine (SVM)
- K-Nearest Neighbors (KNN)
- Random Forest
- Decision Tree
- Logistic Regression
- One-dimensional Convolutional Neural Network (CNN)

All these models are pooled together so as to be tested and compared. The system workflow is to load a network traffic data, preprocess, extract features, train the models etc.

It provides a more agile means of detection by incorporating more than one algorithm into a system. This aids in the comprehension of how either of the models operates and assists in developing a scalable security system to be applied on public Wi-Fi networks.

Attack Categories:

There are 39 types of attacks that represent a set in the dataset that are categorized into four major groups:

- Denial of Service (DoS) – Attacks that flood a network with traffic to stop it from working properly.
- Probe Attacks – Attempts to gather information about network weaknesses through scanning.
- Remote to Local (R2L) – Attacks where a hacker tries to access a local machine from a remote location.
- User to Root (U2R)- Attacks where a user attempts to obtain more privileges within the system i.e. access to root.

VII. Proposed Approach and Workflow

There are seven main stages to propose this methodology:

Stage 1: Data Preparation

The loaded dataset is split in two parts for train and test. The features which act like inputs is converted into labels and used in supervised learning process.

Stage 2: Data Pre-processing

Preprocessing of data guarantees a clean dataset that can be used to train models.

Cleaning

- Cleaning of inconsistencies in the data
- To remove unnecessary queries

Normalization

- Arranging it to a common range
- Avoiding overwhelming of large valued features.
- To ensure stabilization and confluence.

Stage 3: Feature Extraction and Selection

The system will look into the stoppage features according to its behavior. The feature selection is done to:

- Reduce dimensionality
- Remove redundant attributes
- Improve computational efficiency

Stage 4: Classification Models

Multiple algorithms are to be trained and are correlated with each other:

- K-Nearest Neighbors (KNN) – Classifies traffic based on similarity to nearby data points.
- Support Vector Machine (SVM) – Uses most favorable boundaries to group data.
- Random Forest – Having many trees to surpass overfitting.

- Decision Tree – Interpretable classification for decision making.
- Logistic Regression - Light two-way classification.
- Naïve Bayes – Assumes independent features for grouping.
- 1D Convolutional Neural Network (CNN) – Detect the complex patterns in sequential data.

Stage 5: Model Training and Testing

The experimentation is performed on these models to test their generalization.

Stage 6: Evaluation Metrics

- Accuracy – Overall correctness of predictions.
- Precision – Proportion of predicted attacks that are actual.
- Sensitivity (Recall) – Capability to identify real attacks
- Specificity – Capacity to recognize normal traffic properly
- F1-Score – Balances precision and recall.

TABLE II: EVALUATION METRICS

Parameter	Definition	Formula
Accuracy	Measures the overall correctness of the model, how many total predictions were correct.	$\text{Accuracy}=\frac{TP+TN}{TP+TN+FP+FN}$
Precision	Out of all predicted positive cases, how many are actually positive	$\text{Precision}=\frac{TP}{TP+FP}$
Recall	Mean of the differences between predicted and actual values.	$\text{Recall (Sensitivity)}=\frac{TP}{TP+FN}$
Specificity	Measures how well predicted values match actual values (closer to 1 is better).	$\text{Specificity}=\frac{TN}{TN+FP}$

Stage 7: Real-Time Deployment Capability

The last system is forwarded to assist real time detection:

- Timely checking of traffic
- Instant behavior classification
- Creating alerts for suspicious activities.

VIII. EXPERIMENTAL RESULTS

To check if the ML-based IDS works well, many experiments were conducted using different traditional and modern learning models. The goal was to determine its performance in terms of correctness and to identify methods that effectively reduce false alerts in public Wi-Fi settings.

To ensure proper testing of each classifier, they were all trained on the dataset and then evaluated on new, unseen data. The experiment setup needed to reflect real-world conditions that the system would encounter, which allowed for a comparison between normal activity and various attacks.

A. Evaluation Metrics

To assess the system's performance thoroughly, the following measures were used:

- Accuracy: This shows the overall percentage of instances grouped correctly.
- Precision: This tells us how often the system can find an attack when it predicted one.
- Recall (Sensitivity): This measures how well the system can detect actual attacks.
- F1-Score: This is a balance between precision and recall, taking into account both false positives and false negatives.
- ROC-AUC: This shows how well the model states normal and malicious traffic in various thresholds levels.

Logistic Regression

This type of statistical model is commonly used for classification and making predictions. They measure the probability of an event on a set of independent variables. The logit transformation finds the ratio of success to failure. Unlike linear regression which has binary values (0 or 1), the logistic regression has values lying between 0 and 1. It uses the "sigmoid" activation function. It is represented using the formula:

$$h\theta(x) = g(\theta^T x) = 1 / (1 + e^{(-\theta^T x)}) \quad \text{(i)}$$

$$g(z) = 1 / (1 + e^{(-z)}) \quad \text{(ii)}$$

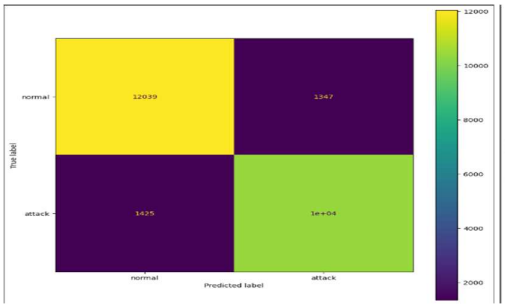

Fig 3. Logistic Regression

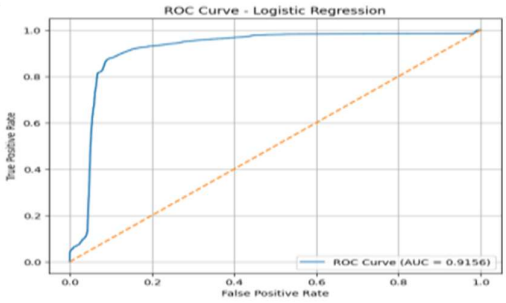

Fig 4: ROC Curve

K Nearest Neighbours

The k-nearest neighbors is supervised non-parametric algorithm using the idea of proximity to group the data points. It assumes that "whenever there is similarity of objects, they are always close to each other".

To determine which data points are closer, we calculate the distance from one point to all other points. This helps to create boundaries create different groups.

It is calculated using Euclidean distance and the formula is given as follows:

$$\text{Distance}(A,B) = \sqrt{[\Sigma(i=1 \text{ to } d)\ (a_i - b_i)^2]} \quad \text{(iii)}$$

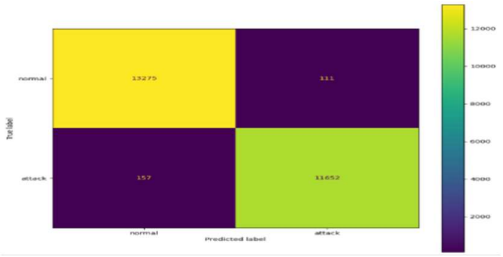

Fig 5. K Neareat Neighbour

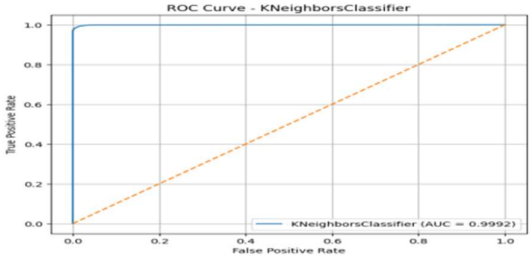

Fig 6: ROC curve

Naïve Bayes

Naïve Bayes classifiers are a set of grouping algorithms based on Bayes' Theorem. This algorithm assumes every feature as independent of each other (one feature does not the another feature) which might not be true but works well. The main advantage of this is that it is overly simplistic and that's why its called as 'naïve'.

Bayes' Theorem

It states that "The probability of A, given the occurance given the B, is equal to the product of the probility of B, given A, and the probability of A divided by the probability of B:

where A and B are events and $P(B) \neq 0$.

- P(A) is the prior probability of A before B is known.
- P(A|B) is the posterier probability of B after B is known.

$$P(A|B) = (P(B|A) \times P(A)) / P(B) \quad \text{(iv)}$$

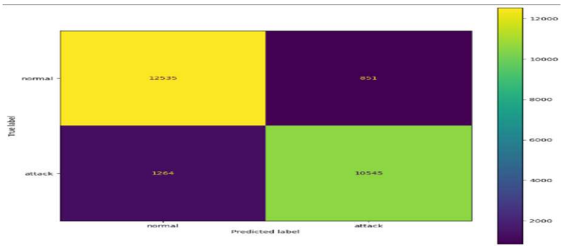

Fig:7 Bayes' Theorem

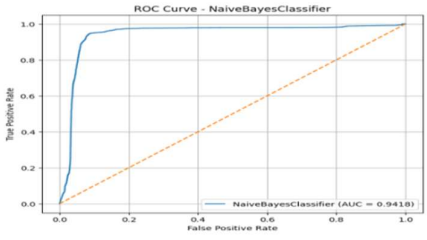

Fig 8: Roc Curve

Support Vector Machines

Support Vector Machine is nothing but finding an optimal hyperplane which maximizes the distance between two classes in an n-dimensional space.

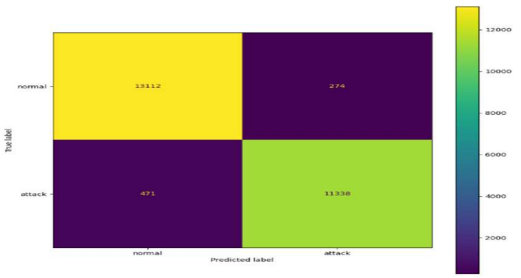

Fig 9: Support Vector machine

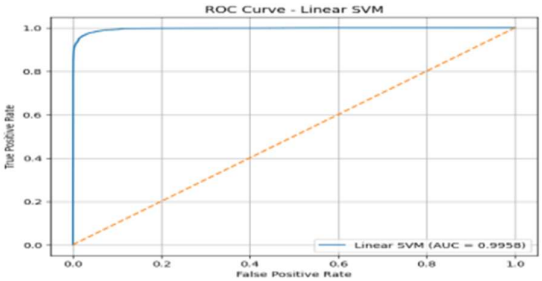

Fig10: ROC curve

Decision Tree

Decision Tree looks like a flowchart, where the internal nodes represent an attribute and the branch of the nodes holds the outcome of that test and each leaf represents a class label.

They group instances by moving down the tree. From the root, it gets split and a new subset is derived having the same value for the objective variable and the process keeps on repeating recursively. While decision trees are very good in making decision making, having too many branches can cause overfitting and it is a time taking process. The best substitute of decision tree is 'Random Forest'.

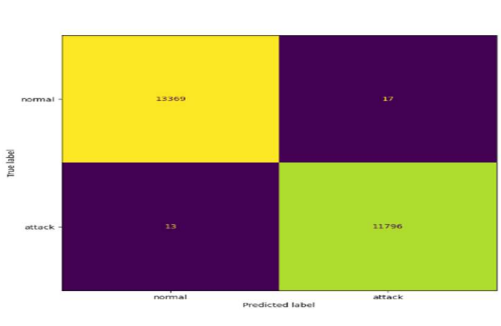

Fig 11: Decision tree

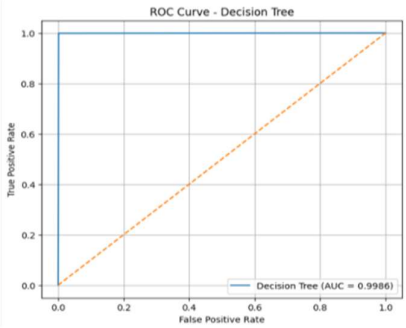

Fig12: ROC curve

TABLE III: MODEL COMPARISON TABLE

Model	Accuracy	Precision	Recall	F1-Score
Logistic Regression	88.85%	88.10%	87.53%	87.80%
KNN	98.94%	99.06%	98.67%	98.86%
Naïve Bayes	91.59%	92.53%	89.30%	90.88%
SVM	97.04%	97.64%	96.01%	96.82%
Decision Tree	99.88%	99.86%	99.89%	99.87%

To ensure model reliability, additional validation methods, including techniques like cross-validation and hyperparameter tuning should be considered.

The Model with the highest accuracy of 99.88% in the above table were the highest score of the Decision Tree model. The reason is that Decision Trees are able to deal with complicated and non-linear associations within the information by dividing the details step by step. But that high level of accuracy can also be due to overfitting particularly when the model is not constrained with parameters such as maximum depth or pruning.

Comparatively, KNN and SVM also scored high with accuracies of 98.94% and 97.04% as their accuracy indicating that they are able to generalize with new data more. The poorer performance of Logistic Regression and Naive Bayes was due to their simple mode of assumptions and the fact that they do not effectively reflect the presence of complex patterns. Thus, although Decision Tree is the most successful one, methods such as pruning, cross-validation, or other similar techniques, such as the Random Forest can be employed to enhance generalization and enhance the effectiveness of the model in practice.

IX. Conclusion

The paper demonstrated a machine learning-driven Intrusion Detection System (IDS) which has been used to secure public Wi-Fi networks. The findings indicate that the Decision Tree model was the most accurate, whereas KNN and SVM also demonstrated high performance levels with enhanced generalization abilities. Whereas high accuracy was obtained, the study emphasizes that overfitting needs to be addressed and model robustness should be ensured to be applied in the real world. Ensuring that the detection performance and scalability are even more advanced, future efforts can be made in ensemble learning methods, like the Random Forest, and deep learning approaches. Altogether, the offered system is efficient and scalable to enhance security in the open Wi-Fi settings.

References

[1] Banakh, R., Piskozub, A., & Opirskyy, I. (2024). Devising a method for detecting evil twin attacks on IEEE 802.11 networks using KNN classification. International Journal of Wireless Security, 12(3), 145–152

[2] Banakh, R., Opirskyy, I., & Piskozub, A. (2023). Data mining approach for evil twin attack identification in Wi-Fi networks. Journal of Network Intelligence, 8(4), 210-219

[3] Salah, Z., & Abu Elsoud, E. (2024). Enhancing intrusion detection in 5G and IoT environments using machine learning on the AWID3 dataset. IEEE Access, 12, 55021-55035. https://doi.org/10.1109/ACCESS.2024.AWID3

[4] Reyes, A., Perdomo, J., & Salazar, R. (2023). A machine learning based two-stage Wi-Fi network intrusion detection system using AWID. Journal of Information Security Research, 14(2), 95-104

[5] Asaduzzaman, M., Majib, M. S., & Rahman, M. M. (2022). Machine learning techniques for efficient Wi-Fi intrusion detection using frame-level features. Wireless Networks, 28(7), 3453-3467

[6] Reyes, A., Perdomo, J., & Ramirez, C. (2023). XAI-enabled two-stage Wi-Fi intrusion detection using AWID dataset. International Journal of Cybersecurity and Digital Forensics, 12(1), 30-41

[7] Da Silva, L. M., Santos, R., & Carvalho, D. (2023). Analysis and identification of evil twin attacks through data science techniques. Procedia Computer Science, 219, 785-792

[8] Kang, G., Lee, J., & Park, H. (2022). Enhancing Wi-Fi reliability using evil twin AP detection based on machine learning. Journal of Information Processing Systems, 18(5), 671-682. https://doi.org/10.3745/JIPS.02.0185

[9] Çetin, B., Daldal, N., & Erdemir, E. (2019). Federated wireless network intrusion detection for public Wi-Fi environments. Proceedings of the 2019 International Conference on Cyber Security, 1-8

[10] Ayoob, K., & Kumar, V. (2025). Intrusion detection in public Wi-Fi networks using SNMP traffic analysis. arXiv Preprint. https://arxiv.org/abs/2501.00411

[11] Ferrag, M. A., Maglaras, L., Moschoyiannis, S., & Janicke, H. (2020). Intrusion detection in cyber security using deep learning: Methodology, data, and a comparison. J. of Information Security and Applications, 50, 102419.

[12] Ahmad, Z., Shahid Khan, A., Shiang, C. W., Abdullah, J., & Ahmad, F. (2021). Machine learning and deep learning approaches to detect intrusion into the network: A systematic study. Emerging Telecommunications Technologies, 32(1), e4150.

[13] Aljawarneh, S., Aldwairi, M., & Yassein, M. B. (2020). Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model. Journal of Computational Science, 25, 152160.

[14] Moustafa, N., & Slay, J. (2021). Assessment of network anomaly detection systems: Statistical analysis of UNSW-NB15 dataset and a comparison with the KDD99 dataset. Information Security Journal, 30(2), 67–80.

[15] Vinayakumar, R., Alazab, M., Soman, K. P., Poornachandran, P., & Al-Nemrat, A. (2020). Smart intrusion detection system based on deep learning method. IEEE Access, 8, 41525–41550.

[16] Khraisat, A., Gondal, I., Vamplew, P., & Kamruzzaman, J. (2021). Overview of intrusion detection systems: Methodologies and data and challenges. Cybersecurity, 4(1), 20.

[17] Kasongo, S. M., & Sun, Y. (2020). A deep learning system filter-based feature engineering based wireless intrusion detection system. IEEE Access, 8, 38597–38607.

[18] Otoum, S., Nayak, A., & Li, D. (2021). The deep-learning-based distributed intrusion detection of IoT systems. Ad Hoc Networks, 110, 102308.

[19] Alsaedi, N., Moustafa, N., Tari, Z., Mahmood, A., & Anwar, A. (2020). Deep learning-based network anomaly detector. The Journal of Network and Computer Applications, 167, 102740.

[20] Kumar, P., Gupta, G. P., & Tripathi, R. (2022). Possible future research direction: Towards design of smart system of cyber attack detection by utilizing hybrid machine learning methods. Future Generation Computer Systems, 124, 261-273.