International journal of basic and applied research

www.pragatipublication.com

ISSN 2249-3352 (P) 2278-0505 (E)

Cosmos Impact Factor-5.86

MCAD: A MACHINE LEARNING BASED CYBERATTACKS DETECTOR IN SOFTWARE-DEFINED NETWORKING (SDN) FOR HEALTHCARE SYSTEMS

P. SIVA PRASAD, Assistance Professor, Dept of MCA, Chirala Engineering College, Chirala, Lakshmiprasad8216@gmail.com

MADDIBOYINA MAHENDRA KUMAR, PG student - MCA, Dept of MCA, Chirala Engineering College, Chirala, mahendrakumarmaddiboyina@gmail.com

Abstract: Significant challenges faced by the healthcare sector in protecting sensitive patient data within software-defined networks (SDNs) are highlighted . With cyber threats becoming increasingly complex, the need for robust security measures in healthcare applications is paramount. The project proposes a Machine Learning-based Cyberattack Detector (MCAD) as a solution. MCAD is designed to use machine learning algorithms to identify and respond to a wide range of cyber threats in healthcare systems. This project addresses the critical importance of advancing cybersecurity measures in healthcare applications. Protecting patient data and ensuring the reliability of healthcare networks are not only critical to safeguarding patient health but also maintaining trust in healthcare institutions. By effectively countering cyber threats and improving network performance, the project seeks to enhance the overall security and resilience of healthcare systems.And also in this project included ensemble methods which are Stacking and Voting Classifiers are implemented to improve the accuracy and they achieved 100% accuracy in cyberattack detection for Healthcare Systems using Software-Defined Networking. Developed a user-friendly Flask-based front end with secure authentication for practical use in healthcare settings.

***Index Terms** - Network resilience, network management, intrusion detection system (IDS), software defined networking, healthcare, machine learning.*

1. INTRODUCTION

In the last few years, SDNs have been extensively used in different fields, principally thanks to their advantages as reliable network technology that allows controlling and managing a network by disaggregating both control and data planes. In contrast to traditional networks, where the network simply has application awareness, the SDN architecture provides additional information about the condition of the entire network from the controller to its applications. Following the recent high-paced progress in information and communications technologies (ICT), healthcare establishments have begun to employ numerous infrastructure factors of the same types of off-the-shelf technologies, applications, and procedures employed by companies from other sectors. This situation was expected, due to the ability of networked or Internet-connected medical tools to increase the effectiveness of asset management, communications, and electronic health records, among other requirements, which reduces cost.Furthermore, the safety of systems and devices, together with user data confidentiality are the two factors that are primarily taken into account in the majority of information systems, since confidentiality and safety are crucial in a healthcare context due to the exacting requirements of the industry. Therefore, it is important that the current McAfee record highlighted that networked medical tools may reveal security gaps in the attempt by the medical industry to incorporate all the technical elements related to networked infrastructure and operational controls though expenses for hospital equipment are expected.

This research aims to enhance the security of healthcare systems by developing a machine learning-based cyber-attack detector (MCAD) implemented within software-defined networks (SDNs). Utilizing a layer three (L3) learning switch application to gather and analyze normal and abnormal network traffic, MCAD will be deployed on the Ryu controller. The study includes extensive testing involving multiple machine learning algorithms and cyberattack scenarios, providing a comprehensive performance evaluation. MCAD demonstrates robust performance with a high F1-score for both normal and attack classes, indicating reliability, while achieving a throughput rate of 5,709,692 samples per second for real-time operations.

The healthcare industry faces a critical challenge in safeguarding sensitive patient data within software-defined networks (SDNs). Despite their advantages, SDNs are susceptible to a wide range of cyber intrusions, endangering network integrity and patient safety. To address this issue, this research aims to develop a machine learning-based cyber-attack detector (MCAD) for healthcare systems, leveraging a layer three (L3) learning switch application on the Ryu controller. This study seeks to comprehensively assess MCAD's performance against various machine learning algorithms and attack scenarios to bolster healthcare data security and network resilience.

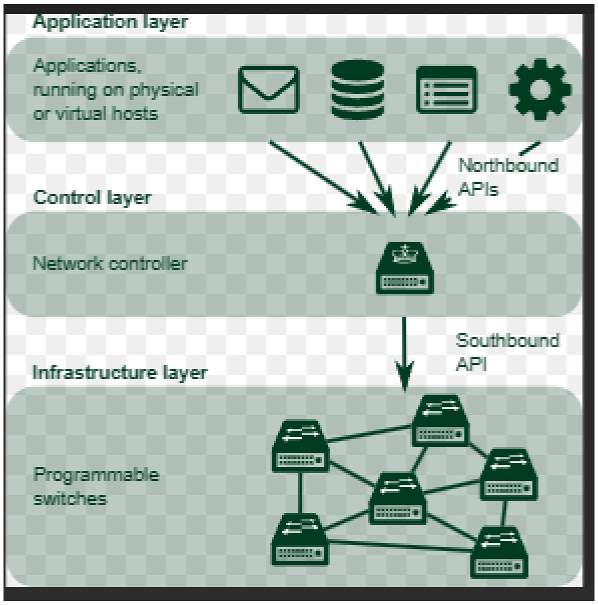


Fig 1 SDN Architecture

Besides the susceptibility of information in healthcare networks, the intricacy, quantity, and variety of tools, particularly networked medical devices (e.g., wireless pacemakers) creating this infrastructure, networks will be exposed to a wider variety of privacy risks and security [4], [5]. During the COVID-19 pandemic, the number of attacks has increased five times. Consequently, 90% of healthcare providers have been subjected to data violations [6]. As proven in recent ransomware incidents [7], the healthcare industry is particularly vulnerable to cyberattacks, which may be attributable to confidentiality breaches (e.g., leaked or comprised sensitive medical records), incidental errors, or deliberate and extensive interference (e.g., caused by flawed construction, use, or function). Researchers have recently begun to explore the prospect of using SDN in healthcare establishments due to the ability of SDN to abstract network policy from network devices [8].

In relation to cyber security in healthcare establishments, SDNs could be employed as protection for medical networks against various harms (e.g., denial-of-service (DoS) and probe attacks). However, in common with current or conventional security resolutions such as intrusion identification and precaution systems or centralized protection methods, SDN solutions do not offer protection to the data and system from insider threats [9]. To illustrate, 92% of healthcare establishments revealed issues faced by their companies due to insider threats and needed appropriate resolutions for protection [10]. This condition makes it important to design functional solutions to reduce insider threats.

2. LITERATURE SURVEY

In present day era use of emerging technologies has given a rise to the healthcare issues. Combination of sensors, the industrial Internet of Things (IIoT), and big data analytics to enhance patient care can lower the healthcare costs. This will enable the patients with more secure, affordable, and rising medical services [8]. Besides problems, such as resource-constrained IoT stuff, identity theft attacks, and malicious insiders, there is a need to address smart healthcare in big data and artificial intelligence using edge computing services. To fix these concerns, we are proposing a software-defined networking (SDN)-based security compliance structure for smart healthcare load migration systems. Toward this end, the use of SDN-IIoT technology for effective and real-time protection against security attacks is being explored by researchers and professionals. In our proposed framework, there are three domains and each domain has one virtual machine and various OpenFlow virtual switches [1,8,12,26,39]. This scenario helps in migrating the heavily loaded domain healthcare data to the lightly loaded domain to make the domain balanced and prevent the migration from happening any type of security attacks. The RYU SDN controller is used to test the simulations and effectiveness of the performance obtained in the mininet after capturing the OpenFlow packets in Wireshark. Secure data management is achieved through the proposed framework and proposed algorithm gives 80% accurate for all the fetched healthcare data packets.

Software defined networks bring many benefits with the centralization, application programmability interfaces and quick implementation of policies across whole network. Scalability and security are improved comparing with traditional networks, but centralized control have some drawbacks as it can be vulnerable for internal or external denial of service attacks. In this article [19], a comparison between two of the most used SDN controllers and the effect of internal denial of service attack towards the southbound interface during switch registration is presented. During the attack the CPU utilization and response time of the controller is collected and analyzed.

This paper shows the implementation of an Intruder Detection System (IDS) integrated into an Artificial Neural Network (ANN), called (Snort + RNA); as an option to mitigate the risks of active computer attacks towards a Software Defined Network (SDN) [20]. Which leverages the network hyperconverged of the data center of the Faculty of Engineering of Applied Science (FICA) at the Technical University of the North. This proposal is tested under the PDCA model offered by the ISO/IEC 27001 standard and the processes provided by the hacker circle. The results show that Snort + RNA detects the anomalies that cause active-type attacks against the SDN, this is visible both in the alerts generated and in the record of the captured traffic, however, it is not possible to analyze all the packets it receives from attacks from DoS since some packages remain on hold or rejected. This shows that, although the system does not evaluate all the packets that circulate on the network, that it takes care of the protection of the SDN, providing alerts when its third parties tried to violate it with attacks that caused an increase in network traffic [12,19,26,28].

Internet of Things (IoT) has emerged as a powerful communication and networking system for smart and automation processing. With the increasing usage of the Internet of Things in numerous critical activities, it is essential to ensure that the communication among these devices is safe and secure. The biggest threat to safe and secure communication is from cyberattacks. Cyberattacks have evolved and become more complex, henceforth posing increased challenges to the data integrity, communication security, and confidentiality of the data. With its success in detecting security vulnerabilities in a communication network, intrusion detection systems are best integrated for securing IoT-based devices [21]. But the integration of an intrusion detection system in an IoT-based network is a challenging task. This paper investigates the state of the art of IoT and intrusion detection system, the technology in use, and the technology challenges by reviewing notable existing works [34]. A systematic literature review of 25 sources comprising 22 research papers and articles covering the threat models, intrusion detection system key challenges in IoT, Proposed models, and implementation of models, reviews, and evaluations are reviewed. The findings explore the needs and the best ways of integrating artificial intelligence-based intrusion detection systems in IoT networks for ensuring security and safety of communication.

Internet of Things (IoT) devices work mainly in wireless mediums; requiring different Intrusion Detection System (IDS) kind of solutions to leverage 802.11 header information for intrusion detection. Wireless-specific traffic features with high information gain are primarily found in data link layers rather than application layers in wired networks. [22] This survey investigates some of the complexities and challenges in deploying wireless IDS in terms of data collection methods, IDS techniques, IDS placement strategies, and traffic data analysis techniques. This paper's main finding highlights the lack of available network traces for training modern machine-learning models against IoT specific intrusions. Specifically, the Knowledge Discovery in Databases (KDD) Cup dataset is reviewed to highlight the design challenges of wireless intrusion detection based on current data attributes and proposed several guidelines to future-proof following traffic capture methods in the wireless network (WN). The paper starts with a review of various intrusion detection techniques, data collection methods and placement methods. [42,44] The main goal of this paper is to study the design challenges of deploying intrusion detection system in a wireless environment. Intrusion detection system deployment in a wireless environment is not as straightforward as in the wired network environment due to the architectural complexities. So this paper reviews the traditional wired intrusion detection deployment methods and discusses how these techniques could be adopted into the wireless environment and also highlights the design challenges in the wireless environment. The main wireless environments to look into would be Wireless Sensor Networks (WSN), Mobile Ad Hoc Networks (MANET) and IoT as this are the future trends and a lot of attacks have been targeted into these networks. So it is very crucial to design an IDS specifically to target on the wireless networks.

3. METHODOLOGY

i) Proposed Work:

The proposed system in the project is a Machine Learning-based Cyberattack Detector (MCAD) specifically designed to enhance the cybersecurity of healthcare systems. It leverages machine learning algorithms to detect and respond to a wide range of cyber threats, safeguarding the sensitive patient data present in healthcare applications and networks. MCAD's adaptability, real-time responsiveness, and comprehensive threat coverage make it an effective solution for countering cyberattacks and improving network security.And added , an ensemble method is implemented that combines the predictive power of individual models, specifically which are Stacking Classifier and a Voting Classifier. Remarkably, both classifiers achieved 100% accuracy, emphasizing the robustness of the ensemble approach in cyberattack detection within Software-Defined Networking for Healthcare Systems[12,14,33]. To further facilitate user testing, we developed a user-friendly front end using the Flask framework. This interface includes user authentication features, ensuring secure access to the Cyberattacks Detector and enhancing the usability of the system in real-world healthcare settings.

ii) System Architecture:

Phase 1: Proposing a Logical Network Topology: The model begins by designing a logical network topology for the healthcare system.

Phase 2: Data Gathering: The model collects data for training and testing the machine learning (ML) model [19,42]. This includes different types of attacks (probe attack, exploit VNC port 5900 remote view vulnerability, and exploit Samba server vulnerability) as well as normal samples.

Phase 3: Data Preprocessing: The collected data is preprocessed to prepare it for training the ML model.

Phase 4: Training and Testing the ML Model: The ML model is trained and tested using various classification algorithms such as KNN, decision tree (DT), random forest (RF), naive Bayes (NB), logistic regression (LR), adaptive boosting (adaboost), and xgboost (XGB). The model constructs a mapping function between inputs and outputs, detecting patterns and minimizing errors. The performance is measured in terms of accuracy [19,42].

Phase 5: Deployment of the project : The trained ML model is deployed on user interface . This allows the model to be implemented in real-time systems, ensuring the overall quality of the healthcare system.

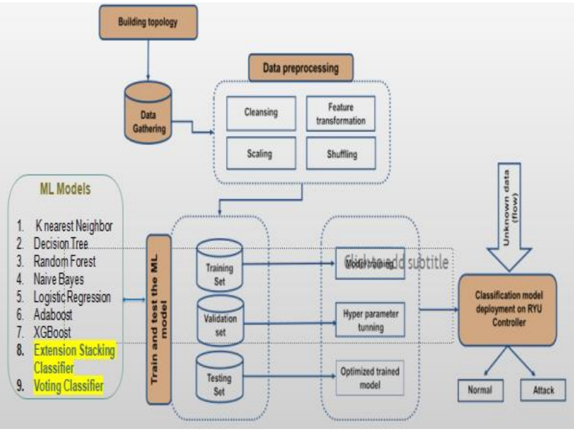
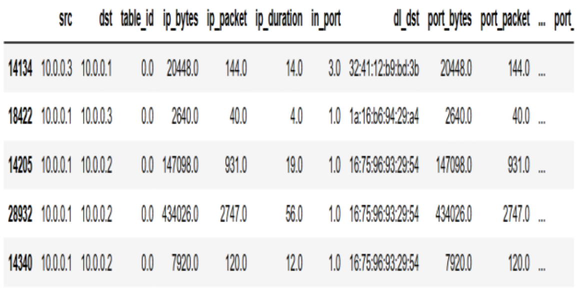


Fig 2 Proposed Architecture

iii) Dataset collection:

MCAD-SDN Dataset: You explore the MCAD-SDN dataset, which likely contains relevant information about network traffic, cyber threats, and other attributes. This step involves gaining an understanding of the dataset's structure, size, and content.



	src	dst	table_id	ip_bytes	ip_packet	ip_duration	in_port	dl_dst	port_bytes	port_packet	...	port_
14134	10.0.0.3	10.0.0.1	0.0	20448.0	144.0	14.0	3.0	32:41:12:b9:bd:3b	20448.0	144.0	...	
18422	10.0.0.1	10.0.0.3	0.0	2640.0	40.0	4.0	1.0	1a:16:b6:94:29:a4	2640.0	40.0	...	
14205	10.0.0.1	10.0.0.2	0.0	147098.0	931.0	19.0	1.0	16:75:96:93:29:54	147098.0	931.0	...	
28932	10.0.0.1	10.0.0.2	0.0	434026.0	2747.0	56.0	1.0	16:75:96:93:29:54	434026.0	2747.0	...	
14340	10.0.0.1	10.0.0.2	0.0	7920.0	120.0	12.0	1.0	16:75:96:93:29:54	7920.0	120.0	...	

Fig 2 MCAD – SDN dataset

iv) Data Processing:

Data processing involves transforming raw data into valuable information for businesses. Generally, data scientists process data, which includes collecting, organizing, cleaning, verifying, analyzing, and converting it into readable formats such as graphs or documents. Data processing can be done using three methods i.e., manual, mechanical, and electronic. The aim is to increase the value of information and facilitate decision-making. This enables businesses to improve their operations and make timely strategic decisions. Automated data processing solutions, such as computer software programming, play a significant role in this. It can help turn large amounts of data, including big data, into meaningful insights for quality management and decision-making.

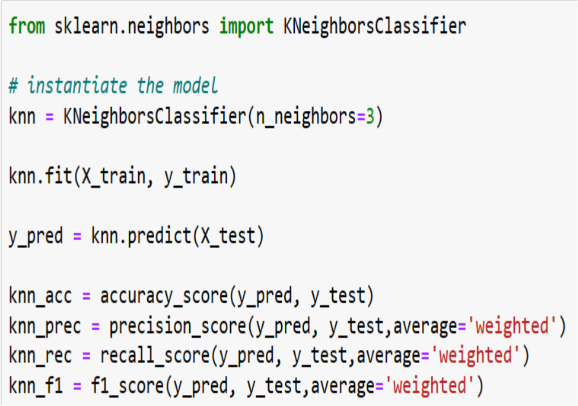
v) Feature selection:

Feature selection is the process of isolating the most consistent, non-redundant, and relevant features to use in model construction. Methodically reducing the size of datasets is important as the size and variety of datasets continue to grow. The main goal of feature selection is to improve the performance of a predictive model and reduce the computational cost of modeling.

Feature selection, one of the main components of feature engineering, is the process of selecting the most important features to input in machine learning algorithms. Feature selection techniques are employed to reduce the number of input variables by eliminating redundant or irrelevant features and narrowing down the set of features to those most relevant to the machine learning model. The main benefits of performing feature selection in advance, rather than letting the machine learning model figure out which features are most important.

vi) Algorithms:

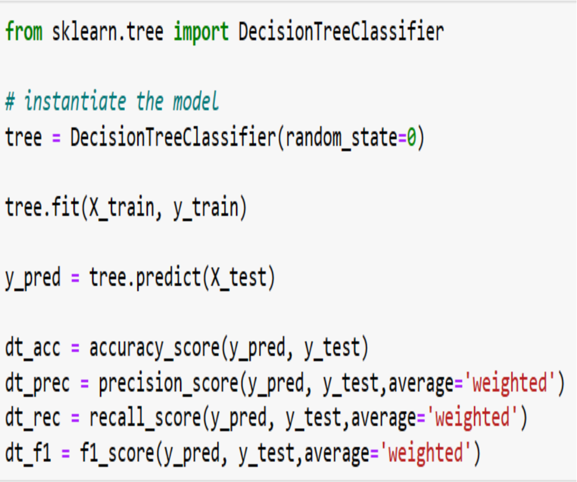
K Nearest Neighbor (KNN) is a supervised algorithm for classification and regression. It classifies data based on the majority class of their k-nearest neighbors (k is user-defined), assuming that similar data points are close in the feature space. KNN can be used to classify network traffic patterns within the healthcare SDN environment [1,8,12]. It helps identify abnormal behavior by comparing patterns to known instances



```
from sklearn.neighbors import KNeighborsClassifier

# instantiate the model
knn = KNeighborsClassifier(n_neighbors=3)

knn.fit(X_train, y_train)

y_pred = knn.predict(X_test)

knn_acc = accuracy_score(y_pred, y_test)
knn_prec = precision_score(y_pred, y_test,average='weighted')
knn_rec = recall_score(y_pred, y_test,average='weighted')
knn_f1 = f1_score(y_pred, y_test,average='weighted')
```

Fig 3 KNN

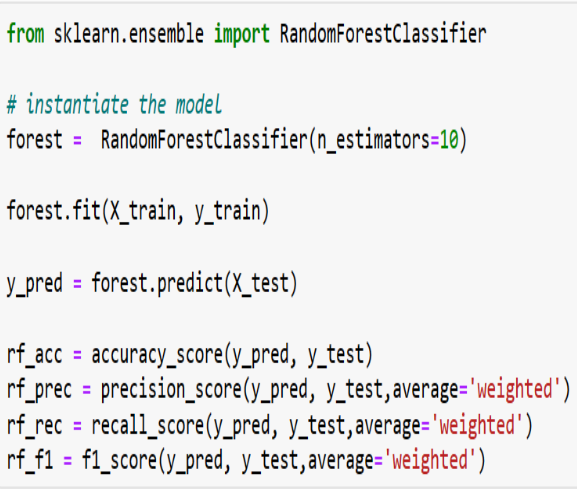
Decision trees are used for classification and regression. They're tree-like structures where nodes are feature tests, and branches lead to outcomes. They make decisions by traversing from the root to leaves using input features Decision trees can be employed to create decision rules for detecting network anomalies. The interpretable nature of decision trees is valuable for understanding the network's behaviour.



```
from sklearn.tree import DecisionTreeClassifier

# instantiate the model
tree = DecisionTreeClassifier(random_state=0)

tree.fit(X_train, y_train)

y_pred = tree.predict(X_test)

dt_acc = accuracy_score(y_pred, y_test)
dt_prec = precision_score(y_pred, y_test,average='weighted')
dt_rec = recall_score(y_pred, y_test,average='weighted')
dt_f1 = f1_score(y_pred, y_test,average='weighted')
```

Fig 4 Decision tree

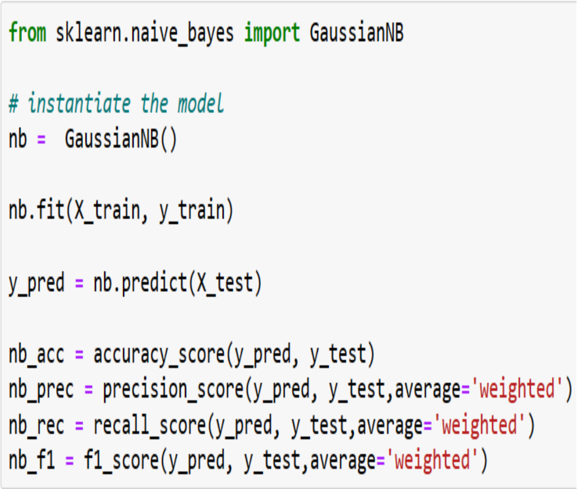
Random Forest is an ensemble method that blends multiple decision trees, forming a forest. Predictions are made by averaging or voting on the trees' predictions. It mitigates overfitting and enhances model accuracy. Random Forest can improve the reliability of cyberattack detection by aggregating predictions from multiple decision trees. It helps mitigate false positives and false negatives in healthcare network security [24], [28], and [30].



```
from sklearn.ensemble import RandomForestClassifier

# instantiate the model
forest =  RandomForestClassifier(n_estimators=10)

forest.fit(X_train, y_train)

y_pred = forest.predict(X_test)

rf_acc = accuracy_score(y_pred, y_test)
rf_prec = precision_score(y_pred, y_test,average='weighted')
rf_rec = recall_score(y_pred, y_test,average='weighted')
rf_f1 = f1_score(y_pred, y_test,average='weighted')
```

Fig 5 Random forest

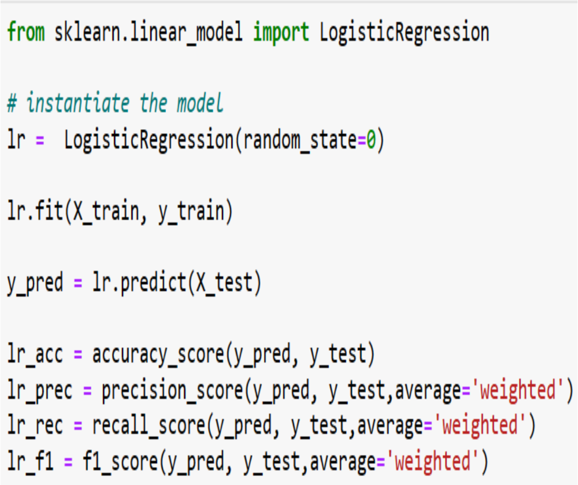
Naive Bayes is a probabilistic classifier using Bayes' theorem. It simplifies by assuming conditional independence between features, often employed in text classification and spam filtering. Naive Bayes can assist in text classification, which is important for detecting malicious traffic in healthcare communication. It's suitable for identifying unusual textual patterns in network data [54].



```
from sklearn.naive_bayes import GaussianNB

# instantiate the model
nb =  GaussianNB()

nb.fit(X_train, y_train)

y_pred = nb.predict(X_test)

nb_acc = accuracy_score(y_pred, y_test)
nb_prec = precision_score(y_pred, y_test,average='weighted')
nb_rec = recall_score(y_pred, y_test,average='weighted')
nb_f1 = f1_score(y_pred, y_test,average='weighted')
```

Fig 6 Naïve bayes

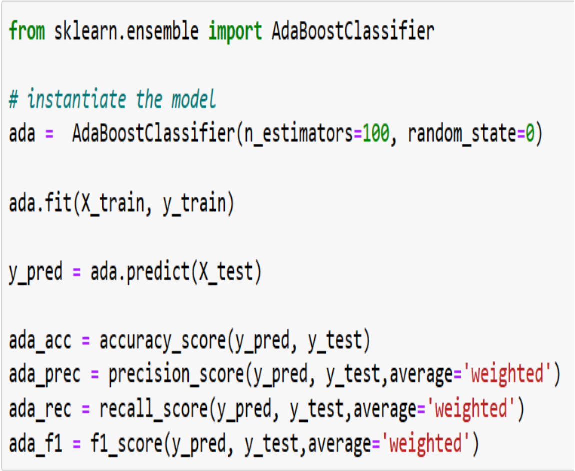
Logistic Regression is a statistical model used for binary classification problems. It estimates the probability that a given input belongs to a particular class. It models the relationship between the dependent variable (binary outcome) and one or more independent variables using the logistic function Logistic Regression can be used to estimate the probability of network events being related to cyberattacks, making it valuable for binary classification in healthcare network security [55].



```
from sklearn.linear_model import LogisticRegression

# instantiate the model
lr =  LogisticRegression(random_state=0)

lr.fit(X_train, y_train)

y_pred = lr.predict(X_test)

lr_acc = accuracy_score(y_pred, y_test)
lr_prec = precision_score(y_pred, y_test,average='weighted')
lr_rec = recall_score(y_pred, y_test,average='weighted')
lr_f1 = f1_score(y_pred, y_test,average='weighted')
```

Fig 7 Logistic regression

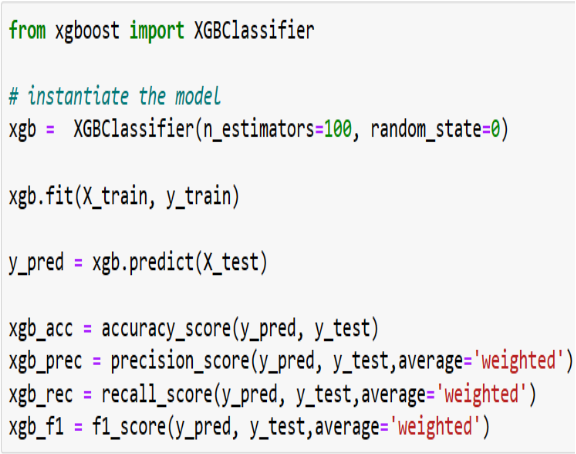
Adaboost is an ensemble method merging weak classifiers to form a strong one. It emphasizes misclassified instances, enabling later classifiers to correct errors. It's commonly used for binary classification. Adaboost can improve the performance of base classifiers, making it a powerful tool for boosting the accuracy of cyberattack detection in healthcare SDNs [56].



```
from sklearn.ensemble import AdaBoostClassifier

# instantiate the model
ada =  AdaBoostClassifier(n_estimators=100, random_state=0)

ada.fit(X_train, y_train)

y_pred = ada.predict(X_test)

ada_acc = accuracy_score(y_pred, y_test)
ada_prec = precision_score(y_pred, y_test,average='weighted')
ada_rec = recall_score(y_pred, y_test,average='weighted')
ada_f1 = f1_score(y_pred, y_test,average='weighted')
```

Fig 8 Adaboost

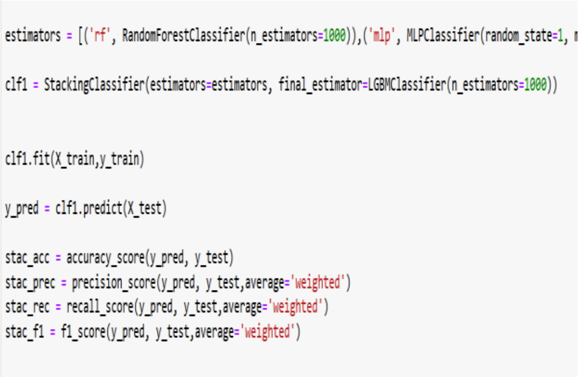
XGBoost is an optimized gradient boosting algorithm for supervised learning, known for its efficiency, accuracy, regularization techniques, handling of missing data, and parallel processing. It's widely favored in machine learning competitions and applications. XGBoost, known for its high accuracy, can be used to build a robust and reliable cyberattack detection model, ensuring the utmost protection for healthcare data.



```
from xgboost import XGBClassifier

# instantiate the model
xgb =  XGBClassifier(n_estimators=100, random_state=0)

xgb.fit(X_train, y_train)

y_pred = xgb.predict(X_test)

xgb_acc = accuracy_score(y_pred, y_test)
xgb_prec = precision_score(y_pred, y_test,average='weighted')
xgb_rec = recall_score(y_pred, y_test,average='weighted')
xgb_f1 = f1_score(y_pred, y_test,average='weighted')
```

Fig 9 XGBoost

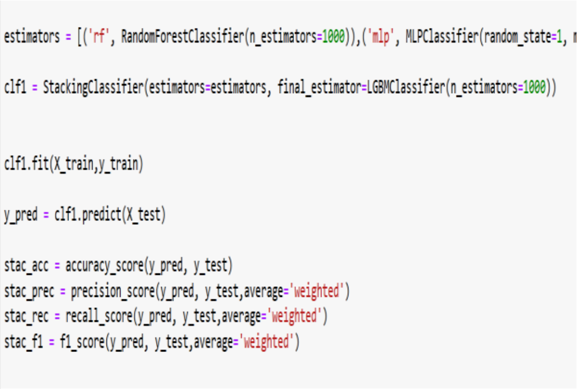
Stacking combines base classifiers to boost predictive performance, utilizing a meta-learner that uses base classifier outputs to make final predictions. It enhances accuracy by capturing diverse patterns. Stacking can be applied to create an ensemble of multiple cyberattack detection models, capturing a wide range of attack patterns and enhancing the overall security of healthcare systems



```
estimators = [('rf', RandomForestClassifier(n_estimators=1000)),('mlp', MLPClassifier(random_state=1,

clf1 = StackingClassifier(estimators=estimators, final_estimator=LGBMClassifier(n_estimators=1000))


clf1.fit(X_train,y_train)

y_pred = clf1.predict(X_test)

stac_acc = accuracy_score(y_pred, y_test)
stac_prec = precision_score(y_pred, y_test,average='weighted')
stac_rec = recall_score(y_pred, y_test,average='weighted')
stac_f1 = f1_score(y_pred, y_test,average='weighted')
```

Fig 10 Stacking classifier

Voting is an ensemble method that unifies predictions from multiple base classifiers. It can be hard (majority vote) or soft (class probabilities). Voting classifiers enhance model robustness and accuracy by leveraging multiple models' strengths. A voting classifier can be implemented to combine the decisions of multiple detection models, allowing for more reliable and robust identification of cyberattacks in the healthcare network.



```
estimators = [('rf', RandomForestClassifier(n_estimators=1000)),('mlp', MLPClassifier(random_state=1, m

clf1 = StackingClassifier(estimators=estimators, final_estimator=LGBMClassifier(n_estimators=1000))


clf1.fit(X_train,y_train)

y_pred = clf1.predict(X_test)

stac_acc = accuracy_score(y_pred, y_test)
stac_prec = precision_score(y_pred, y_test,average='weighted')
stac_rec = recall_score(y_pred, y_test,average='weighted')
stac_f1 = f1_score(y_pred, y_test,average='weighted')
```

Fig 11 Voting classifier

4. EXPERIMENTAL RESULTS

Precision: Precision evaluates the fraction of correctly classified instances or samples among the ones classified as positives. Thus, the formula to calculate the precision is given by:

Precision = True positives/ (True positives + False positives) = TP/(TP + FP)

$$\text{Precision} = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

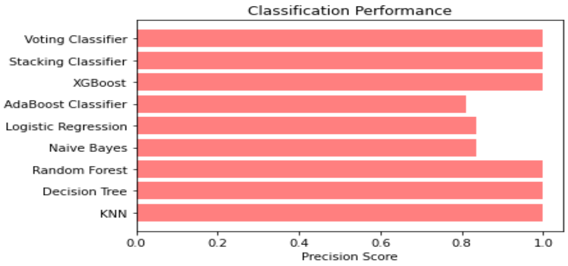


Fig 6 Precision comparison graph

Recall: Recall is a metric in machine learning that measures the ability of a model to identify all relevant instances of a particular class. It is the ratio of correctly predicted positive observations to the total actual positives, providing insights into a model's completeness in capturing instances of a given class.

$$Recall = \frac{TP}{TP + FN}$$

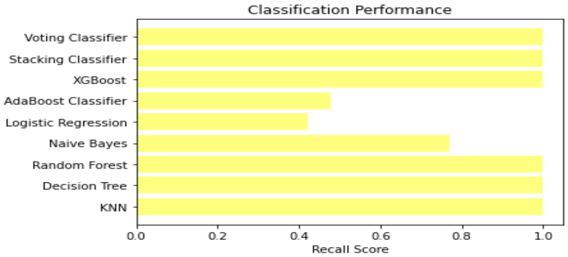


Fig 7 Recall comparison graph

Accuracy: Accuracy is the proportion of correct predictions in a classification task, measuring the overall correctness of a model's predictions.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

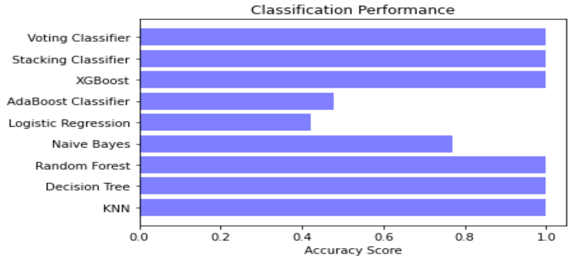


Fig 8 Accuracy graph

F1 Score: The F1 Score is the harmonic mean of precision and recall, offering a balanced measure that considers both false positives and false negatives, making it suitable for imbalanced datasets.

$$\text{F1 Score} = 2 * \frac{\text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} * 100$$

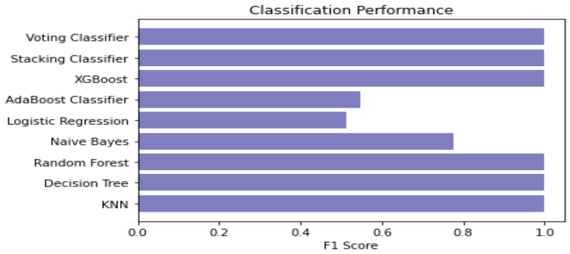
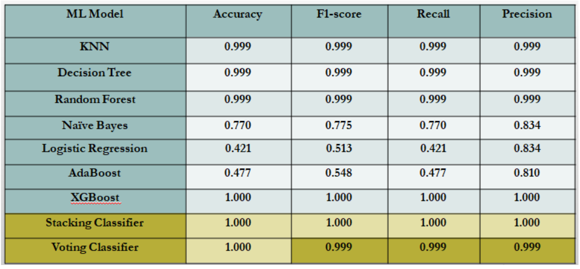
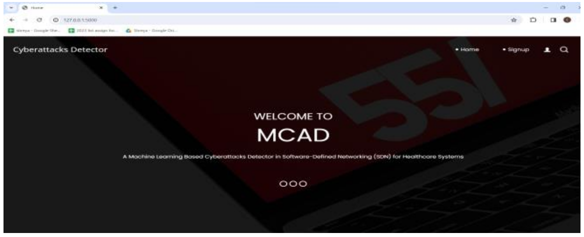


Fig 9 F1Score



ML Model	Accuracy	F1-score	Recall	Precision
KNN	0.999	0.999	0.999	0.999
Decision Tree	0.999	0.999	0.999	0.999
Random Forest	0.999	0.999	0.999	0.999
Naïve Bayes	0.770	0.775	0.770	0.834
Logistic Regression	0.421	0.513	0.421	0.834
AdaBoost	0.477	0.548	0.477	0.810
XGBoost	1.000	1.000	1.000	1.000
Stacking Classifier	1.000	1.000	1.000	1.000
Voting Classifier	1.000	0.999	0.999	0.999

Fig 10 Performance Evaluation



WELCOME TO

MCAD

A Machine Learning Based Cyberattacks Detector in Software-Defined Networking (SDN) for Healthcare Systems

Fig 11 Home page

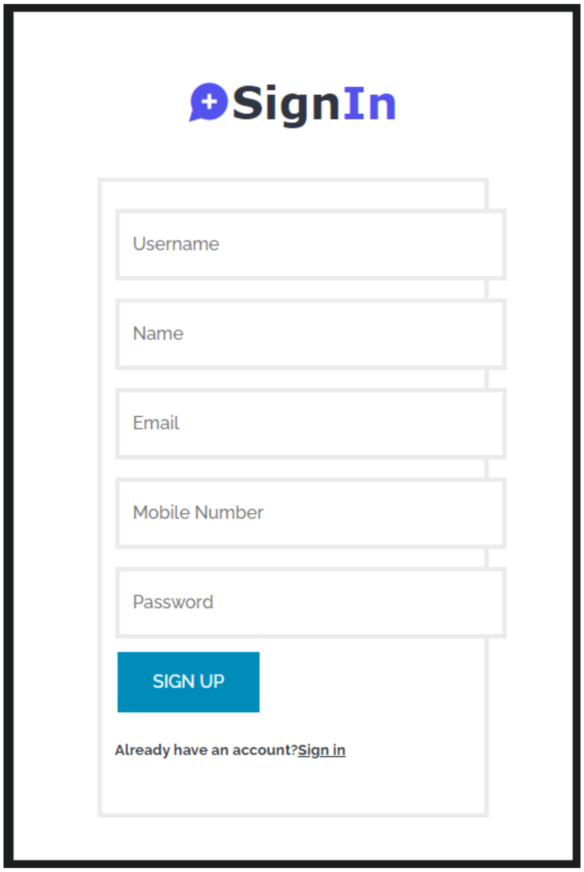
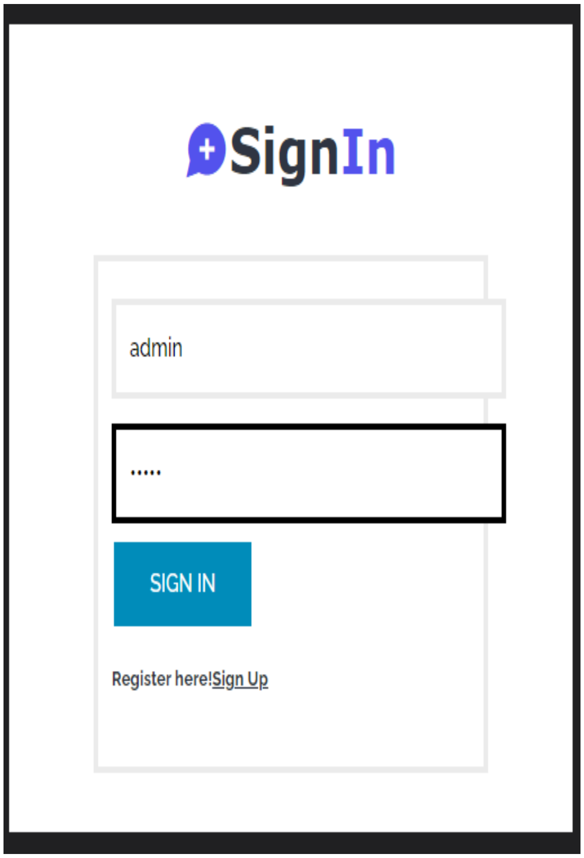


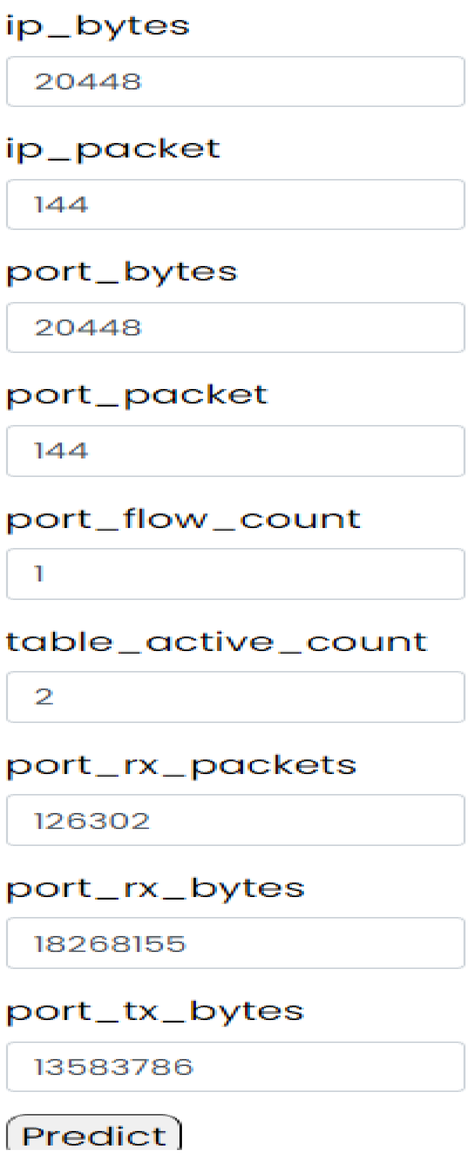
Fig 12 Signin page



admin

•••••

Fig 13 Login page



ip_bytes
20448

ip_packet
144

port_bytes
20448

port_packet
144

port_flow_count
1

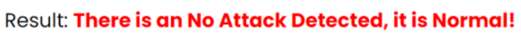
table_active_count
2

port_rx_packets
126302

port_rx_bytes
18268155

port_tx_bytes
13583786

Fig 14 User input



Result: There is an No Attack Detected, it is Normal!

Fig 15 Predict result for given input

5. CONCLUSION

The project has successfully engineered a robust cyberattack detection system leveraging the power of machine learning techniques, contributing to enhanced cybersecurity. We conducted an in-depth exploration of the MCAD-SDN dataset, undertaking essential data preprocessing tasks such as feature selection and encoding, ensuring the dataset's readiness for analysis. In our quest for an effective cyberattack detection solution, we rigorously assessed various machine learning models, including ensemble methods, to measure their accuracy and suitability for detecting cyberattacks. Among the array of models considered,The successful implementation and outstanding performance of the ensemble algorithm, Stacking and Voting Classifiers with a 100% accuracy rate, underscore its robustness and efficacy as an advanced cyberattack detection solution for securing healthcare Software-Defined Networking systems [37]. This project marks a significant step forward in bolstering cybersecurity measures and defending against evolving threats in the digital landscape.

6. FUTURE SCOPE

Further research can be conducted to explore the application of the machine learning-based cyberattack detector (MCAD) in other sectors beyond healthcare, such as finance, transportation, or critical infrastructure, to enhance their security against cyber threats [35,37,42]. The performance of the MCAD can be evaluated and optimized by testing it with a larger and more diverse dataset of both normal and attack traffic, as well as different machine learning algorithms. Ongoing development and improvement of the MCAD can focus on enhancing its real-time capabilities, scalability, and adaptability to evolving cyber threats. Collaboration with industry stakeholders, cybersecurity experts, and regulatory bodies can help in the implementation and standardization of the MCAD in healthcare systems and other critical sectors.

REFERENCES

[1] M. Jarschel, T. Zinner, T. Hossfeld, P. Tran-Gia, and W. Kellerer, ‘‘Interfaces, attributes, and use cases: A compass for SDN,’’ IEEE Commun. Mag., vol. 52, no. 6, pp. 210–217, Jun. 2014.

[2] W. Meng, K.-K.-R. Choo, S. Furnell, A. V. Vasilakos, and C. W. Probst, ‘‘Towards Bayesian-based trust management for insider attacks in healthcare software-defined networks,’’ IEEE Trans. Netw. Service Manage., vol. 15, no. 2, pp. 761–773, Jun. 2018.

[3] J. T. Kelly, K. L. Campbell, E. Gong, and P. Scuffham, ‘‘The Internet of Things: Impact and implications for health care delivery,’’ J. Med. Internet Res., vol. 22, p. 11, Nov. 2020.

[4] (2022). Networked Medical Devices: Security and Privacy Threats—Sym antec—[PDF Document]. [Online]. Available: https://fdocuments. net/document/networked-medical-devices-security-and-privacy-threatssymantec.html

[5] P. A. Williams and A. J. Woodward, ‘‘Cybersecurity vulnerabilities in medical devices: A complex environment and multifaceted problem,’’ Med. Devices, Evidence Res., vol. 8, pp. 305–316, Jul. 2015.

[6] C. M. Williams, R. Chaturvedi, and K. Chakravarthy, ‘‘Cybersecurity risks in a pandemic,’’ J. Med. Internet Res., vol. 22, no. 9, Sep. 2020, Art. no. e23692.

[7] N. Thamer and R. Alubady, ‘‘A survey of ransomware attacks for healthcare systems: Risks, challenges, solutions and opportunity of research,’’ in Proc. 1st Babylon Int. Conf. Inf. Technol. Sci. (BICITS), I. Babil, Ed., Apr. 2021, pp. 210–216.

[8] H. Babbar, S. Rani, and S. A. AlQahtani, ‘‘Intelligent edge load migration in SDN-IIoT for smart healthcare,’’ IEEE Trans. Ind. Informat., vol. 18, no. 11, pp. 8058–8064, Nov. 2022.

[9] R. Hasan, S. Zawoad, S. Noor, M. M. Haque, and D. Burke, ‘‘How secure is the healthcare network from insider attacks? An audit guideline for vulnerability analysis,’’ in Proc. IEEE 40th Annu. Comput. Softw. Appl. Conf. (COMPSAC), Jun. 2016, pp. 417–422.

[10] (Apr. 2015). 92% of Healthcare IT Admins Fear Insider Threats Thales. Accessed: Mar. 21, 2023. [Online]. Available: https://cpl.thalesgroup.com/about-us/newsroom/news-releases/92-healthcare-it-admins-fearinsider-threats

[11] D. Chaulagain, K. Pudashine, R. Paudyal, S. Mishra, and S. Shakya, ‘‘OpenFlow-based dynamic traffic distribution in software-defined networks,’’ in Mobile Computing and Sustainable Informatics. Singapore: Springer, Jul. 2021, pp. 259–272.

[12] R. Khondoker, A. Zaalouk, R. Marx, and K. Bayarou, ‘‘Feature-based comparison and selection of software defined networking (SDN) controllers,’’ in Proc. World Congr. Comput. Appl. Inf. Syst. (WCCAIS), Jan. 2014, pp. 1–7.

[13] T. Mekki, I. Jabri, A. Rachedi, and L. Chaari, ‘‘Software-defined networking in vehicular networks: A survey,’’ Trans. Emerg. Telecommun. Technol., vol. 33, no. 10, pp. 1–10, Apr. 2021, doi: 10.1002/ett.4265.

[14] Z. Ghaffar, A. Alshahrani, M. Fayaz, A. M. Alghamdi, and J. Gwak, ‘‘A topical review on machine learning, software defined networking, Internet of Things applications: Research limitations and challenges,’’ Electronics, vol. 10, no. 8, p. 880, Apr. 2021, doi: 10.3390/electronics10080880.

[15] C.-S. Li and W. Liao, ‘‘Software defined networks [guest editorial],’’ IEEE Commun. Mag., vol. 51, no. 2, p. 113, Feb. 2013.

[16] M. H. Rehmani, A. Davy, B. Jennings, and C. Assi, ‘‘Software defined networks-based smart grid communication: A comprehensive survey,’’ IEEE Commun. Surveys Tuts., vol. 21, no. 3, pp. 2637–2670, 3rd Quart., 2019.

[17] L. F. Eliyan and R. Di Pietro, ‘‘DoS and DDoS attacks in software defined networks: A survey of existing solutions and research challenges,’’ Future Gener. Comput. Syst., vol. 122, pp. 149–171, Sep. 2021, doi: 10.1016/j.future.2021.03.011.

[18] K. Benton, L. J. Camp, and C. Small, ‘‘OpenFlow vulnerability assessment,’’ in Proc. 2nd ACM SIGCOMM Workshop Hot Topics Softw. Defined Netw., 2013, pp. 151–152, doi: 10.1145/2491185.2491222.

[19] B. Mladenov and G. Iliev, ‘‘Studying the effect of internal DOS attacks over SDN controller during switch registration process,’’ in Proc. Int. Symp.

Netw., Comput. Commun. (ISNCC), Jul. 2022, pp. 1–4.

[20] H. Domínguez-Limaico, W. N. Quilca, M. Zambrano, F. Cuzme-Rodríguez, and E. Maya-Olalla, ‘‘Intruder detection system based artificial neural network for software defined network,’’ in Proc. Int. Conf. Technol. Res. Cham, Switzerland: Springer, Aug. 2022, pp. 315–328.

[21] S. A. Mehdi and S. Z. Hussain, ‘‘Survey on intrusion detection system in IoT network,’’ in Proc. Int. Conf. Innov. Comput. Commun. Singapore: Springer, Sep. 2022, pp. 721–732.

[22] V. Ponnusamy, M. Humayun, N. Z. Jhanjhi, A. Yichiet, and M. F. Almufareh, ‘‘Intrusion detection systems in Internet of Things and mobile ad-hoc networks,’’ Comput. Syst. Sci. Eng., vol. 40, no. 3, pp. 1199–1215, 2022, doi: 10.32604/csse.2022.018518.

[23] K. Malasri and L. Wang, ‘‘Securing wireless implantable devices for healthcare: Ideas and challenges,’’ IEEE Commun. Mag., vol. 47, no. 7, pp. 74–80, Jul. 2009.

[24] D. Yin, L. Zhang, and K. Yang, ‘‘A DDoS attack detection and mitigation with software-defined Internet of Things framework,’’ IEEE Access, vol. 6, pp. 24694–24705, 2018.

[25] R. Wang, Z. Jia, and L. Ju, ‘‘An entropy-based distributed DDoS detection mechanism in software-defined networking,’’ in Proc. IEEE Trustcom/BigDataSE/ISPA, vol. 1, Aug. 2015, pp. 310–317.

[26] S. M. Mousavi and M. St-Hilaire, ‘‘Early detection of DDoS attacks against SDN controllers,’’ in Proc. Int. Conf. Comput., Netw. Commun. (ICNC), Feb. 2015, pp. 77–81.

[27] S. Murtuza and K. Asawa, ‘‘Mitigation and detection of DDoS attacks in software defined networks,’’ in Proc. 11th Int. Conf. Contemp. Comput., Aug. 2018, pp. 1–3.

[28] X. You, Y. Feng, and K. Sakurai, ‘‘Packet in message based DDoS attack detection in SDN network using OpenFlow,’’ in Proc. 5th Int. Symp. Comput. Netw. (CANDAR), Nov. 2017, pp. 522–528.

[29] S. Y. Mehr and B. Ramamurthy, ‘‘An SVM based DDoS attack detection method for Ryu SDN controller,’’ in Proc. 15th Int. Conf. Emerg. Netw. Exp. Technol., New York, NY, USA, Dec. 2019, pp. 72–73, doi: 10.1145/3360468.3368183.

[30] Q. Niyaz, W. Sun, and A. Y. Javaid, ‘‘A deep learning based DDoS detection system in software-defined networking (SDN),’’ ICST Trans. Secur. Saf., vol. 4, no. 12, Dec. 2017, Art. no. 153515.

[Online]. Available: https://publications.eai.eu/index.php/sesa/article/view/211

[31] G. Lucky, F. Jjunju, and A. Marshall, ''A lightweight decision-tree algorithm for detecting DDoS flooding attacks,'' in Proc. IEEE 20th Int. Conf. Softw. Quality Rel. Secur. Companion (QRS-C), Dec. 2020, pp. 382–389.

[32] J. Ye, X. Cheng, J. Zhu, L. Feng, and L. Song, ''A DDoS attack detection method based on SVM in software defined network,'' Secur. Commun. Netw., vol. 2018, pp. 1–8, Jan. 2018.

[33] P. Radoglou-Grammatikis, K. Rompolos, P. Sarigiannidis, V. Argyriou, T. Lagkas, A. Sarigiannidis, S. Goudos, and S. Wan, ''Modeling, detecting, and mitigating threats against industrial healthcare systems: A combined software defined networking and reinforcement learning approach,'' IEEE Trans. Ind. Informat., vol. 18, no. 3, pp. 2041–2052, Mar. 2022.

[34] T. A. S. Srinivas and S. S. Manivannan, ''Prevention of hello flood attack in IoT using combination of deep learning with improved rider optimization algorithm,'' Comput. Commun., vol. 163, pp. 162–175, Nov. 2020.

[35] A. Kanavalli, A. Gupta, A. Pattanaik, and S. Agarwal, ''Realtime DDoS detection and mitigation in software defined networks using machine learning techniques,'' Int. J. Comput., vol. 10, pp. 353–359, Sep. 2022. [Online]. Available: https://computingonline.net/ computing/article/view/2691

[36] A. Erfan, ''DDoS attack detection scheme using hybrid ensemble learning and ga algorithm for Internet of Things,'' PalArch's J. Archaeol. Egypt/Egyptol., vol. 18, no. 18, pp. 521–546, Jan. 2022. [Online]. Available: https://archives.palarch.nl/index.php/jae/article/view/10546

[37] Y. K. Saheed and M. O. Arowolo, ''Efficient cyber attack detection on the Internet of Medical Things-smart environment based on deep recurrent neural network and machine learning algorithms,'' IEEE Access, vol. 9, pp. 161546–161554, 2021.

[38] A. H. Celdrán, K. K. Karmakar, F. Gómez Mármol, and V. Varadharajan, ''Detecting and mitigating cyberattacks using software defined networks for integrated clinical environments,'' Peer-Peer Netw. Appl., vol. 14, no. 5, pp. 2719–2734, Sep. 2021.

[39] M. S. Elsayed, N.-A. Le-Khac, and A. D. Jurcut, ''InSDN: A novel SDN intrusion dataset,'' IEEE Access, vol. 8, pp. 165263–165284, 2020.

[40] X. Cai, K. Shi, K. She, S. Zhong, Y. Soh, and Y. Yu, ‘‘Performance error estimation and elastic integral event triggering mechanism design for T–S fuzzy networked control system under dos attacks,’’ IEEE Trans. Fuzzy Syst., vol. 31, no. 4, pp. 1–12, Apr. 2023.

[41] X. Cai, K. Shi, K. She, S. Zhong, and Y. Tang, ‘‘Quantized sampled-data control tactic for T–S fuzzy NCS under stochastic cyber-attacks and its application to truck-trailer system,’’ IEEE Trans. Veh. Technol., vol. 71, no. 7, pp. 7023–7032, Jul. 2022.

[42] A. O. Alzahrani and M. J. F. Alenazi, ‘‘ML-IDSDN: Machine learning based intrusion detection system for software-defined network,’’ Concurrency Comput., Pract. Exper., vol. 35, no. 1, pp. 1–12, Jan. 2023.

[43] K. S. Bhosale, M. Nenova, and G. Iliev, ‘‘The distributed denial of service attacks (DDoS) prevention mechanisms on application layer,’’ in Proc. 13th Int. Conf. Adv. Technol., Syst. Services Telecommun. (TELSIKS), Oct. 2017, pp. 136–139.

[44] A. Almazyad, L. Halman, and A. Alsaeed, ‘‘Probe attack detection using an improved intrusion detection system,’’ Comput., Mater. Continua, vol. 74, no. 3, pp. 4769–4784, 2023, doi: 10.32604/cmc.2023.033382.

[45] A. Sadeghian, M. Zamani, and S. M. Abdullah, ‘‘A taxonomy of SQL injection attacks,’’ in Proc. Int. Conf. Informat. Creative Multimedia, Sep. 2013, pp. 269–273.

[46] L. Zhang-Kennedy, S. Chiasson, and R. Biddle, ‘‘Password advice shouldn’t be boring: Visualizing password guessing attacks,’’ in Proc. APWG eCrime Researchers Summit, Sep. 2013, pp. 1–11.

[47] Z. Su and G. Wassermann, ‘‘The essence of command injection attacks in web applications,’’ ACM SIGPLAN Notices, vol. 41, no. 1, pp. 372–382, Jan. 2006.

[48] M. Pivarníková, P. Sokol, and T. Bajtoš, ‘‘Early-stage detection of cyber attacks,’’ Information, vol. 11, no. 12, p. 560, Nov. 2020.

[49] K. V. A. Reddy, S. R. Ambati, Y. S. R. Reddy, and A. N. Reddy, ‘‘AdaBoost for Parkinson’s disease detection using robust scaler and SFS from acoustic features,’’ in Proc. Smart Technol., Commun. Robot. (STCR), Oct. 2021, pp. 1–6.

[50] I. T. Jolliffe and J. Cadima, ‘‘Principal component analysis: A review and recent developments,’’ Philos. Trans. Roy. Soc. A, Math., Phys. Eng. Sci., vol. 374, Apr. 2016, Art. no. 20150202, doi: 10.1098/rsta.2015.0202.

[51] P. Cunningham and S. J. Delany, ‘‘K-nearest neighbour classifiers: 2nd edition (with Python examples),’’ 2020, arXiv:2004.04523.

[52] E. H. Sussenguth, ‘‘An algorithm for automatic design of logical cryogenic circuits,’’ IEEE Trans. Electron. Comput., vol. EC-10, no. 4, pp. 623–630, Dec. 1961.

[53] P. H. Swain and H. Hauska, ‘‘The decision tree classifier: Design and potential,’’ IEEE Trans. Geosci. Electron., vol. GE-15, no. 3, pp. 142–147, Jul. 1977.

[54] Y. Ji, S. Yu, and Y. Zhang, ‘‘A novel Naive Bayes model: Packaged hidden Naive Bayes,’’ in Proc. 6th IEEE Joint Int. Inf. Technol. Artif. Intell. Conf., Aug. 2011, pp. 484–487.

[55] X. Zou, Y. Hu, Z. Tian, and K. Shen, ‘‘Logistic regression model optimization and case analysis,’’ in Proc. IEEE 7th Int. Conf. Comput. Sci. Netw. Technol. (ICCSNT), Oct. 2019, pp. 135–139.

[56] Y. Freund and R. E. Schapire, ‘‘A decision-theoretic generalization of on-line learning and an application to boosting,’’ J. Comput. Syst. Sci., vol. 55, pp. 119–139, Aug. 1995. [Online]. Available: https://www. sciencedirect.com/science/article/pii/S002200009791504X

[57] T. Chen and C. Guestrin, ‘‘XGBoost: A scalable tree boosting system,’’ 2016, arXiv:1603.02754.