MACHINE LEARNING BASED CYBER THREAT DETECTION FOR HEALTHCARE SYSTEMS

[1]Aloney Neeraj, [2]Dr. V. Uma Rani, [3]Dr. Sunitha Vanamala

[1] Student, Department of Information Technology, University College of Engineering Science and Technology, JNTU, Kukatpally, Hyderabad.

[2] Professor Of CSE, Department of Information Technology, University College of Engineering Science and Technology, JNTU, Kukatpally, Hyderabad.

[3] Lecturer, Department of Computer Science, TSWRDCW, Warangal East, Warangal, Telangana, India.

Abstract: The healthcare industry has huge obstructions with regards to shielding private patient data on software-defined networks (SDNs). Solid safety efforts are critical for medical care applications in light of the fact that digital assaults are getting more modern. The proposed arrangement is a Cyberattack Detector (MCAD) that depends on Machine Learning. MCAD is made to perceive and respond to an assortment of cyberthreats in medical care frameworks by using ML methods. The crucial meaning of further developing network protection shields in healthcare applications is tended to by this review. Defending patient wellbeing and maintaining patient confidence in medical care organizations rely upon safeguarding patient information and ensuring the constancy of medical services organizations. The venture means to further develop network execution and alleviate digital assaults to fortify the general security and strength of healthcare systems. Furthermore, the review utilized troupe methods including stacking and casting a ballot classifiers to increment accuracy. They utilized programming characterized systems administration to distinguish cyberattacks on healthcare systems with 100 percent accuracy. made a front end utilizing Flask that is not difficult to involve and has safe confirmation for use in certifiable healthcare settings.

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

1. INTRODUCTION

Lately, SDNs have been generally utilized in a few regions because of their dependability and capacity to control and oversee networks by disaggregating control and information planes. Not at all like conventional networks, which just have application mindfulness, SDN configuration offers additional organization status data from the regulator to its applications. Following the fast headway of information and communications technologies (ICT), medical care establishments are utilizing numerous infrastructural parts of off-the-rack innovation, applications, and methodology utilized by different associations. This was anticipated since arranged or Web associated clinical instruments further develop resource the executives, correspondences, and electronic wellbeing records, diminishing costs.Since privacy and security are significant in healthcare because of the business' severe prerequisites, most data frameworks focus on framework and gadget wellbeing and client information secrecy. In spite of the fact that clinic gear costs are normal, the ongoing McAfee record noticed that organized clinical apparatuses may uncover security holes in the clinical business' endeavor to consolidate all specialized components connected with arranged foundation and functional controls.

This venture fosters a machine learning-based cyber-attack detector (MCAD) for programming characterized organizations to further develop healthcare system security. MCAD will be carried out on the Ryu regulator utilizing a L3 learning change application to evaluate typical and strange organization traffic. An itemized presentation assessment is given by assessing a few ML strategies and cyberattack situations. MCAD's solid F1-score for both typical and assault classes demonstrates trustworthiness, and its constant throughput rate is 5,709,692 examples each second.

Safeguarding delicate patient information in programming characterized networks is a main issue for medical care. Notwithstanding their advantages, SDNs are defenseless against a few cyberattacks that undermine network trustworthiness and patient wellbeing. This exploration utilizes a layer three (L3) learning change application on the Ryu regulator to develop an machine learning-based cyber-attack detector (MCAD) for medical care frameworks. MCAD's exhibition against ML calculations and assault situations will be assessed in this undertaking to further develop healthcare data security and network flexibility.

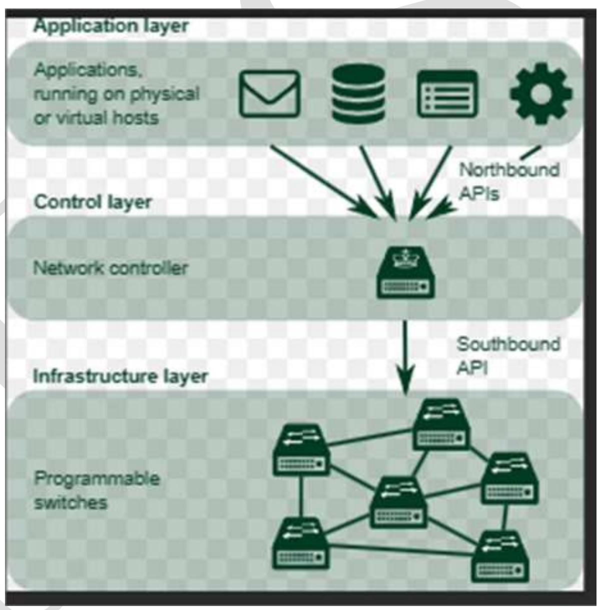

Fig 1 SDN Architecture

Notwithstanding the weakness of data in healthcare networks, the unpredictability, amount, and variety of instruments, quite arranged medical devices (e.g., remote pacemakers), constructing this foundation will expand protection and security dangers [4], [5]. Assaults have fivefold ascended during the Coronavirus pandemic. Information breaks have impacted 90% of medical services suppliers [6]. As shown by ongoing ransomware occurrences [7], the medical services industry is especially defenseless against cyberattacks because of secrecy breaks (e.g., released or involved delicate clinical records), accidental blunders, or intentional and broad obstruction.

SDN's capacity to isolate network strategy from network gadgets has driven analysts to consider using it in medical services [8].

SDNs could protect clinical organizations from pernicious attacks like denial-of-service (DoS) and testing assaults. SDN arrangements, such intrusion detection and prevention systems and brought together assurance draws near, don't shield information and frameworks against insider dangers [9]. For example, 92% of medical services organizations revealed insider danger chances and required security [10]. To moderate insider risks, useful arrangements are required.

2. LITERATURE SURVEY

Arising innovation have expanded medical services difficulties today. Sensors, IIoT, and large information investigation can work on understanding consideration and cut medical care costs. This will give patients more secure, less expensive, and higher clinical consideration [8]. Notwithstanding asset compelled IoT, wholesale fraud, and threatening insiders, brilliant medical care in enormous information and man-made brainpower need edge registering administrations. We propose a SDN-based security consistence structure for brilliant medical services load relocation frameworks to resolve these issues. Scientists and specialists are exploring SDN-IIoT advancements for continuous security insurance. Three spaces with one virtual machine and numerous OpenFlow virtual changes make around our system [1,8,12,26,39]. This situation adjusts the space by moving medical services information from the completely stacked area to the gently stacked area, forestalling security attacks. The RYU SDN regulator recreates and assesses mininet execution after Wireshark catches OpenFlow bundles. System and calculation give secure information dealing with and 80% precision for all procured medical services information parcels.

Centralization, application programmability points of interaction, and quick approach execution across entire organizations are advantages of programming characterized networks. Adaptability and security are superior to conventional organizations, albeit incorporated control may be defenseless against DDoS attacks. In [19], two well known SDN regulators are analyzed and the impact of inside refusal of administration assault on the southward connection point during switch enlistment is inspected. Regulator central processor use and response time are considered during the attack.

In this review, a Intruder Detection System (IDS) coordinated into a Artificial Neural Network (ANN) (Snort+RNA) is introduced to decrease the gamble of dynamic PC attacks on a Software Defined Network (SDN) [20]. Which utilizes the Technical University of the North Faculty of Engineering of Applied Science (FICA) server farm's hyperconverged network. The ISO/IEC 27001 PDCA model and hacking circle strategies are utilized to test this thought. Grunt + RNA recognizes peculiarities causing dynamic sort assaults on SDN, as found in cautions and traffic records. In any case, a few bundles stay on hold or dismissed, restricting examination of DoS assaults. This shows that, while the framework doesn't evaluate each organization parcels, it safeguards the SDN by cautioning outsiders when they attempt to break it with attacks that increment network traffic [12,19,26,28].

IoT is a refined correspondence and systems administration innovation for brilliant and mechanized handling. With the Web of Things being utilized in additional imperative assignments, free from even a hint of harm gadget network is critical. Cyberattacks represent the most serious risk to get correspondence. Cyberattacks have gotten progressively confounded, compromising information honesty, correspondence security, and mystery. Intrusion detection systems are brilliant for IoT gadget security since they recognize correspondence network security defects [21]. Nonetheless, incorporating an IDS into an IoT network is troublesome. This study audits major IoT and interruption identification framework endeavors to evaluate the best in class, innovation, and challenges [34]. An extensive writing investigation of 25 sources incorporates 22 examination papers and articles on danger models, IoT IDS center issues, proposed models, execution, surveys, and assessments. The discoveries look at the requests and best practices for coordinating AI-based IDS in IoT networks to get correspondence.

Most of Internet of Things (IoT) gadgets utilize remote means, requiring various IDS frameworks to involve 802.11 header data for interruption identification. Information joins, not application layers, in wired networks have remote explicit traffic qualities with significant data gain. [22] This survey analyzes remote IDS arrangement issues in information gathering, IDS strategies, area, and traffic information handling. Absence of organization follows for preparing contemporary AI models against IoT interruptions is this paper's key outcome. In light of current information properties, the Knowledge Discovery in Databases (KDD) Cup dataset is assessed to feature remote interruption location configuration issues and propose various rules to future-confirmation remote organization traffic catch draws near. Intrusion detection, data collecting, and situation strategies are assessed to begin the article. [42,44] The plan issues of remote IDS are the focal point of this examination. Remote IDS execution is more muddled attributable to structural contrasts. This paper examines wired interruption location sending strategies, talks about how they might be utilized remotely, and addresses remote plan issues. Wireless Sensor Networks (WSN), Mobile Ad Hoc Networks (MANET), and IoT are future improvements that have been focused on for attacks. Consequently, remote organization explicit IDS design is fundamental.

3. METHODOLOGY

i) Proposed Work:

The undertaking's recommended innovation is a Machine Learning-based Cyberattack Detector (MCAD), made particularly to further develop healthcare systems' cybersecurity. It utilizes ML calculations to recognize and neutralize an assortment of cyberthreats, safeguarding the confidential patient data found in healthcare networks and applications. As a result of its adaptability, fast reaction time, and broad danger inclusion, MCAD is a valuable device for foiling cyberattacks and upgrading network security. Moreover, an ensemble approach — that is, a voting classifier and a stacking classifier — is utilized to consolidate the prescient capacity of independent models. The way that the two classifiers surprisingly achieved 100 percent accuracy features the strength of the ensemble method in Software-Defined Networking for Healthcare Systems cyberattack detection [12,14, 33]. We made an easy to understand front end with the Flask framework to assist with client testing. This connection point's client verification highlights give safe admittance to the Cyberattacks Detector and work on the framework's convenience in real healthcare conditions.

ii) System Architecture:

Phase 1: Propose logical network topology: The model starts by creating a logical network topology for the healthcare system.

Phase 2: Gather information: To train and evaluate the machine learning (ML) model, the model collects data [19, 42]. These include standard examples and other forms of attacks (probe attacks, exploitation of the remote view vulnerability on VNC port 5900, and exploitation of the Samba server vulnerability).

Phase 3: Data Preprocessing: The collected data is preprocessed in preparation for training the ML model.

Phase 4: Training and Testing the ML Model: Several classification techniques such as Logistic Regression (LR), Naive Bayes (NB), Decision Tree (DT), Random Forest (RF), Adaptive Boosting (Adaboost), and xgboost (XGB) are used to train and evaluate the machine learning model. By constructing a mapping function between inputs and outputs, the model minimizes errors and looks for patterns. Accuracy is used to measure performance [19, 42].

Phase 5: Deployment of the project: The user interface uses the trained machine learning model. This allows the model to be applied in a real-time system, ensuring the overall quality of the healthcare system.

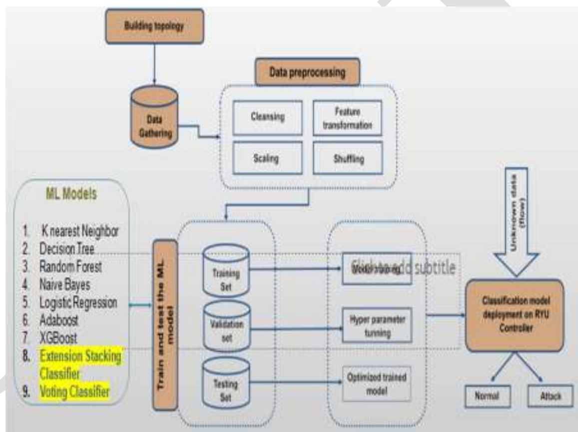

Fig 2 Proposed Architecture

iii) Dataset collection:

MCAD-SDN Dataset: You inspect the MCAD-SDN dataset, which most likely remembers critical subtleties for cyberthreats, network traffic, and different qualities. Grasping the sum, sythesis, and design of the dataset is the objective of this stage.

	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 dataset

iv) Data Processing:

Data processing transforms raw information into business-helpful data. Information researchers accumulate, sort out, clean, check, break down, and orchestrate information into diagrams or papers. Data can be handled physically, precisely, or electronically. Data ought to be more significant and decision-production simpler. Organizations might upgrade activities and settle on basic decisions quicker. PC programming improvement and other mechanized information handling innovations add to this. Big data can be transformed into significant bits of knowledge for quality administration and independent direction.

v) Feature selection:

Feature selection chooses the most steady, non-repetitive, and pertinent elements for model turn of events. As data sets extend in amount and assortment, purposefully bringing down their size is significant. The fundamental reason for feature selection is to increment prescient model execution and limit processing cost.

One of the vital pieces of feature engineering is picking the main attributes for machine learning algorithms. To diminish input factors, feature selection methodologies take out copy or superfluous elements and limit the assortment to those generally critical to the ML model. Rather than permitting the ML model pick the main qualities, feature selection ahead of time enjoys a few benefits.

vi) Algorithms:

K Nearest Neighbor (KNN) is a supervised regression and classification techniques. Assuming that comparable data points are close to each other in the feature space, data is classified according to a majority class of k-nearest neighbors, where k is user-defined. In SDN healthcare settings, ANNs can be used to classify network traffic patterns [1, 8, 12]. Comparing patterns to known occurrences can help identify anomalous behavior.

```
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

Regression and classification are done through **decision trees**. These are tree-like structures where branches lead to outcomes and nodes act as functional tests. They move from root to leaf using input features to make decisions. You can use decision trees to create decision rules to identify anomalies in the network. The interpretable structure of decision trees helps you understand the behavior of the network.

```python
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 a group technique that combines many decision trees to create a forest. Predictions are made by averaging or adjusting the predictions made by the trees. This improves the accuracy of the model and reduces overfitting. Random forests can improve the accuracy of cyber-attack detection by combining predictions from multiple decision trees. From the perspective of healthcare network security, it helps reduce false positives and false negatives [24], [28], [30].

```python
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 classifiers based on Bayes' theorem. The task is made easier by assuming conditional independence between features, a technique commonly used in spam filtering and text classification. Text classification is a key task for identifying malicious traffic in medical communications, where Naive Bayes can help. It can be applied to network data to detect anomalous text patterns [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 for binary classification problems. It calculates the probability that an input is classified into a particular class. A logistic function is used to represent the relationship between a dependent variable (a binary outcome) and one or more independent factors. Logistic regression is useful for binary classification in healthcare network security, as it can be used to calculate the probability that a network event is related to a cyber-attack [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 a group technique that combines weak classifiers to create a stronger classifier. It highlights misclassified cases so that subsequent classifiers can correct the errors. It is often used in binary classification. Adaboost is an effective approach to improve the accuracy of cyber-attack detection in healthcare SDNs, as it can improve the performance of basic classifiers [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 improved gradient boosting approach to supervised learning known for its effectiveness, accuracy, missing data handling, regularization strategies, and parallel processing. It is a popular choice for machine learning applications and competitions. Due to its outstanding accuracy, XGBoost can be used to create powerful and reliable cyber-attack detection models that ensure the highest level of security for medical 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 Improve prediction performance by combining base classifiers and using a meta-learner to generate a final prediction based on the output of the base classifiers. Capturing different patterns improves accuracy. Stacking allows many cyber attack detection models to be combined into one ensemble, which can detect different attack patterns and improve the overall security of the healthcare system.

```
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 10 Stacking classifier

Voting is a group method that combines predictions from many base classifiers. It can be soft (class probability) or rigid (majority vote). Voting classifiers combine the advantages of many models, improving the robustness and accuracy of the model. The ability to combine the decisions of many detection models using voting classifiers makes the identification of cyber attacks in healthcare networks more robust and reliable.

```
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 estimates the level of positive cases or tests precisely sorted. Precision is determined utilizing the recipe:

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

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

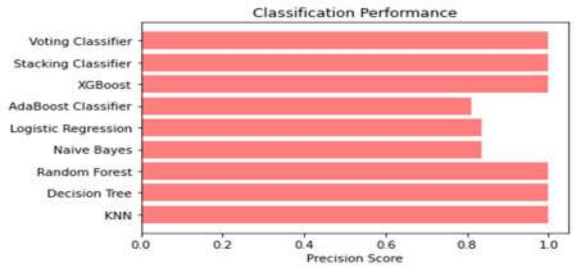

Fig 6 Precision comparison graph

Recall: Recall in machine learning evaluates the model's ability to recognize all significant examples of a class. It indicates the model's performance in detecting events in a class by accurately comparing expected positive recognitions with perfectly positive recognitions.

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

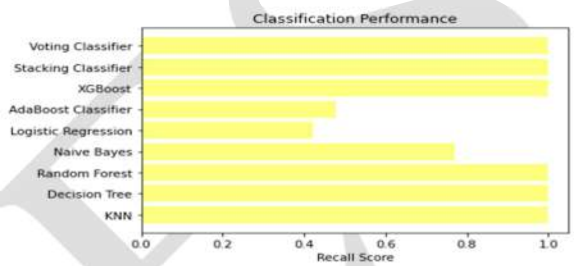

Fig 7 Recall comparison graph

Accuracy: A test's accuracy is its ability to recognize debilitated from sound cases. To quantify test accuracy, figure the small part of true positive and true negative in completely broke down cases. Numerically, this is:

Accuracy = TP + TN TP + TN + FP + FN.

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

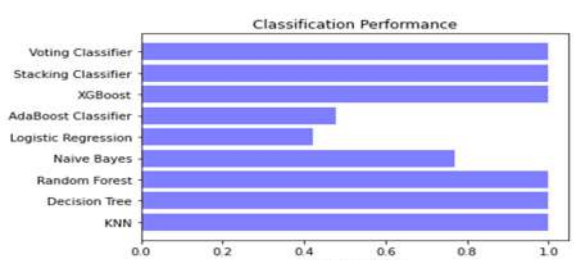

Fig 8 Accuracy graph

F1 Score: Machine learning model accuracy is estimated by F1 score. Combining precision and recall for a model. The precision measure estimates how often a model makes correct predictions across an entire dataset.

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

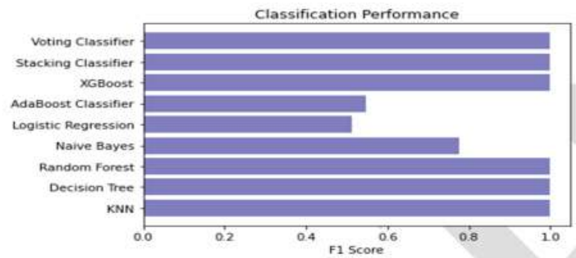

Fig 9 F1Score

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

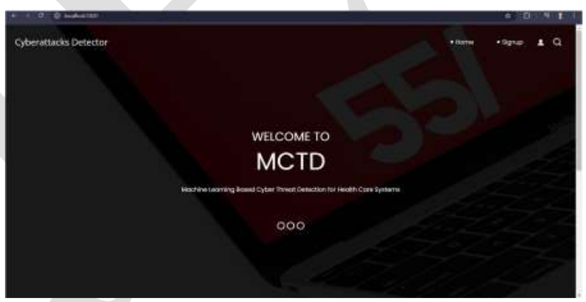

Fig 11 Home page

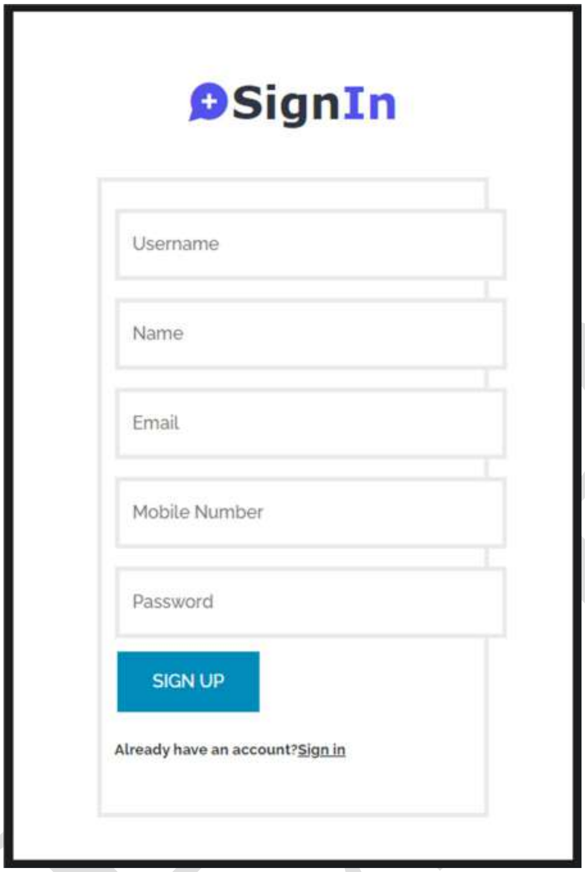

Fig 12 Signin page

Fig 13 Login page

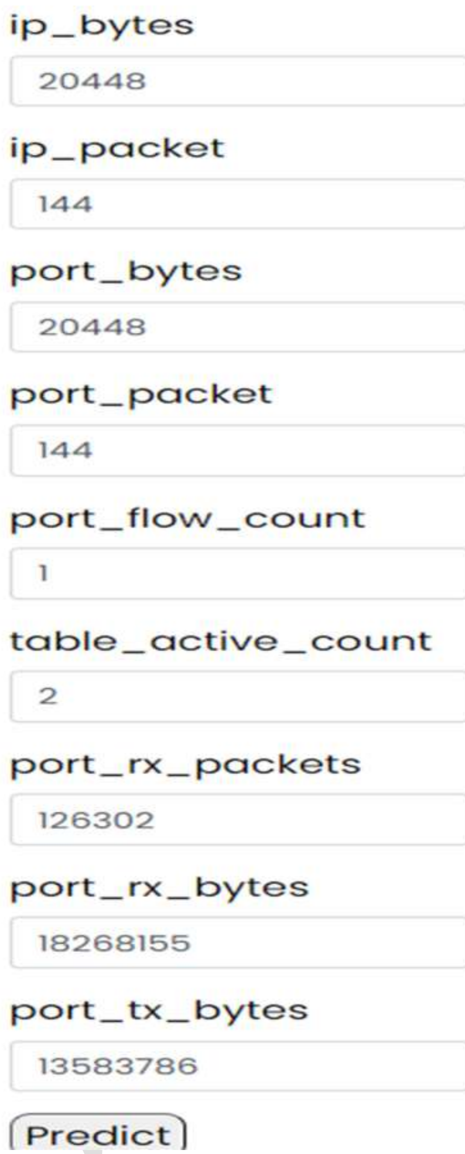

Fig 14 User input

Fig 15 Predict result for given input

5. CONCLUSION

By really using ML strategies to construct major areas of strength for a detection system, the exploration has further developed cybersecurity. We completely analyzed the MCAD-SDN dataset, performing vital information readiness tasks such feature selection and encoding to ensure the dataset was ready for investigation. We completely assessed a scope of ML models, including group draws near, as we continued looking for a proficient cyberattack detection arrangement, to measure their accuracy and relevance. The ensemble calculation's surprising presentation, including Stacking and Voting Classifiers with a 100 percent accuracy rate, and its effective execution among the range of models thought about feature its heartiness and viability as a high level cyberattack detection answer for shielding medical services Software-Defined Networking systems [37]. With regards to fortifying cybersecurity and battling off evolving cyberthreats, this drive addresses a significant progression.

6. FUTURE SCOPE

To improve cyber security in enterprises other than healthcare, like banking, transportation, and basic foundation, the ML based cyberattack detector (MCAD) can be examined [35,37,42]. Test the MCAD with a greater and more expanded dataset of typical and assault traffic and other ML methods to evaluate and upgrade its exhibition. The MCAD might be created to expand its constant abilities, versatility, and digital danger flexibility. Industry partners, network protection experts, and administrative associations might help take on and normalize the MCAD in healthcare and other key businesses.

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.