AN ENHANCED ENSEMBLE CLASSIFIERS FOR CYBERCRIMES DETECTION IN FINANCIAL NETWORKS

Abraham, Danlami
abdanabraham@gmail.com
Departmentt of Computer Science

Garba, Etemi Joshua
e.j.garba@mau.edu.ng
Departmentt of Computer Science

Malgwi, Yusuf Musa
malgwi@mau.edu.ng
Departmentt of Computer Science

Dogo, Siyani Ezra
siyani.elisha@gmail.com
Departmentt of Computer Science

Abstract

The growing reliance on the use of financial networks worldwide has resulted in great concern for cyber security. Financial network is well-connected network that allows for financial transaction and free from errors, it connects people worldwide to interact, share content, and engage in discussions of mutual interest that know no geographical boundaries. Financial setups are the target for the bots due to the financial assets, sensitive data, and data collection methods. The existing security techniques such as machine learning (ML) classifiers, multi-factor authentication, and penetration tester are not proactive enough to detect attacks in financial network platform. The aim of this work is to develop cybercrimes detection model for detection of botnet attacks in financial network. The Specific objectives of the thesis were to: Design a cybercrimes detection model incorporating advanced machine learning algorithms specifically tailored for detecting botnet activities in financial networks. Optimize a user interface model and Evaluate the performance of the new model compare viz-a-viz best existing machine language classifiers and state-of-the-art intrusion detection systems. Google Colab, Draw.io, Python and the Unified Modeling Language (UML) were used to design the model and evaluated using accuracy, precision, detection rate, F1 Score, and Receiver Operating Characteristics (ROC) Curve using a generated dataset of 2,830,742 sizes on CIC-IDS2017 Dataset from simulated networks. The result of the ensemble model includes Classifier 99.98%, 99.98%, 99.98%, 99.98%, and 99.22% respectively.

Keywords: Machine Learning, Deep Learning, *financial networks*, IDS, Cybercrimes, Security Network

Introduction

The security of financial networks is of utmost importance to both banking institutions and their clients in an increasingly digital environment where financial transactions are carried out electronically. The digitalization of financial services has increased convenience like never before, with a wide range of internet services like electronic funds transfers, mobile banking, and automated teller machines, the financial industry is a leader in technical innovation, but it has also left these networks vulnerable to a wide range of developing cyber threats (Hanaa and Sultan 2023). They have introduced vulnerabilities that bad actors are eager to take advantage of the pervasiveness of cybercrimes, which can range from sophisticated nation-state-sponsored operations to opportunistic cybercriminal activity, highlights the urgent need for cutting-edge and adaptable security solutions in the financial industry. Financial networks have become top targets for hostile actors looking to acquire illegal access, money, or disrupt financial activities due to their quantity of valuable financial data and assets. Cybercriminals target financial networks repeatedly in order to obtain unauthorized access, steal sensitive data, and commit

financial fraud. Therefore, it has never been more important to protect the availability, integrity, and confidentiality of banking services. Financial institutions must constantly innovate and adapt to secure their assets and keep their clients' trust as attackers become more sophisticated (Buba *et al*., 2020).

This research acknowledges that, despite their importance, the conventional cybersecurity measures used by financial institutions, such as firewalls and intrusion detection systems, are frequently insufficient to counter the constantly changing strategies of cyber adversaries. So this research sets out to provide a security system that is especially suited to the special difficulties and demands of financial networks in response to the rising threat landscape, and sets out on a crucial trip to tackle this urgent problem by suggesting the creation of a financial network security system that makes use of cyber security networks (Calvin *et al.,* 2021).

An enhanced ensemble models are deception-based cybersecurity that uses the use of bogus systems and resources to entice, find, and examine harmful activity in financial network. As a useful tool in the cybersecurity field, an enhanced ensemble models have attracted attention for their ability to spot new threats, comprehend attack methods, and improve incident response skills. With its proactive and deceptive approach to cybersecurity, an enhanced ensemble models presents a viable way to improve the security posture of financial networks. Assailants are drawn into a controlled environment via an enhanced ensemble classifiers, which are intended to imitate real network assets and give important information about their strategies, techniques, and motivations

The major problems identified from the existing studies includes: The activation of the recommended protection features may come long after the assault has taken place, potentially devastating the targeted user. The users' sensitive information like details related to credit card or ATM card can be accessed and used to defraud the user as the Rule Based Access Controls has no mechanism to secure the bank network account with compromised credentials. Botnets continue to lead the rat-race of inventing new attacking strategies while security experts continue to follow behind with a reactive security mechanism. (Buba *et al*., 2020) There is need to explore an enhanced ensemble models for cybercrimes in financial Network. An enhanced ensemble models are built to monitor, detect and reduce the activities of cybercrimes in financial network.

Cybercrime Attacks

Cybercrime attack includes Data theft, massive DDoS (distributed denial of service) assaults, bulk spam and phishing email campaigns are all examples of botnet attacks. Botnet assaults happen when an attacker has gained control of a significant number of computers. Due to the vast amount of information that Cybercrime attacks may access, they are frequently directed on large-scale groups and organizations. The programmers may take control of innumerable devices with this attack and use them to further their cunning goals. Owners of botnets may reach thousands of PCs at once and command them to do malicious tasks. Bots initially gain access to these devices by attacking the security frameworks of the PCs using special Trojan infections (Hanaa and Sultan 2023). They then carry out command and control programming to

enable them to achieve malicious activities in bank network Calvin *et al* (2021). These drills may be automated to support as many synchronized attacks as would be wise. Several types of botnet attacks can consist of the following gains:

I. Botnet attacks result may lead to unexpected application downtime.
II. Authorizing plans of leaked credentials (certificate stuffing attacks) that lead to account takeovers
III. Attacks against web applications to steal costumers credentials data and bank resource
IV. Granting the attacker access to the device and its affiliation with the company or bank network

Sometimes, attackers will grant access to the botnet network, often referred to as the "zombie" organization, so that other cybercriminals can use it for their own nefarious activities, such as launching a spam campaign.

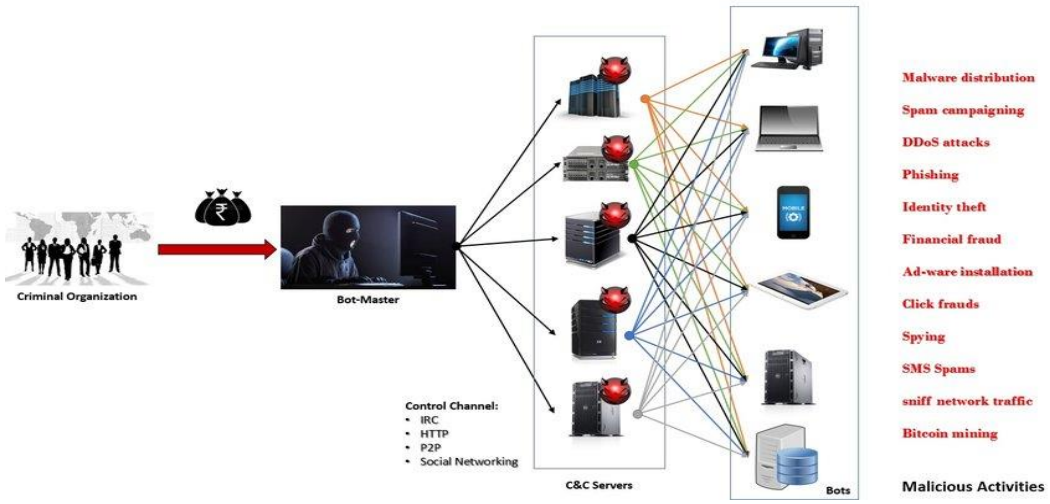

Figure 1: Cybercrime Attacks (Buba *et al*., 2020)

THE RELATED LITERATURE REVIEWED

The following are included in the research review

Author/Year	Methodology	Main Contribution's	Research Gap
Buba *et al*., (2020)	NB, DT, & RF	Developed Machine Learning-Based Botnet Detection System	Detections of bot were based on invalid accounts.
Delplace *et al*. (2020)	SVM, NB, RF & Neural Network	Developed a Solution Based approaches to Detects Malware traffic in a Network	Detected attacks based on the abnormality of the Network
Najla et al., 2020	DT-SVMNB classifier	Uses multipronged approach to Report trends, ZeuS and its variants	Focus was on reporting vulnerable users l networks not cybercrimes

Tyrone & Gubler, (2020)	(RF,SVM,DT, Naive Bayes), Neural Network.	They offered the White hat Bot (WHB), a cutting-edge defensive tactic with recorded tests and outcomes that help in defending bank networks	Detection features were based on fake accounts and anti-viruses not bots
Rafal et al., (2020)	RF, SVM, Decision Tree, Naive Bayes Classifier	Construct an experimental environment that allows alertness and Suspension of botnet activities	Detection features were based on fake user accounts not cybercrime
Damenu & Beaumont, 2020	Long Shortterm Memory Neural Networks	The study provide platform for easily accessible and secure network connections.	Prone to over fitting and it takes longer time to train
Zhao *et al*., (2020)	The use of Assemble REP Classifier	Developed a novel method for detecting botnet activity focused on traffic behavior	The detection approaches were based on harmful and normal attribute
Calvin et al., (2021)	RF, SVM, DT, Naive Bayes, Neural Network	Theory provides strong support for phishing defenses with 91.8% accuracy.	Ineffective executive coordination and security awareness
Calleja *et al.,* (2021)	DT, LR,KNN & NB	Developed Bank Network system to detect botnet assaults	The model detected base on the customers behavior
Dimitis et al., (2021)	Logistic Regression & SVM	Frameworks that detect newly created botnets and their traffic patterns have been built	Degradation in performance when exposed to large dataset
Hung et al., 2021	Anomaly- based	Detecting potentially infected bots by using machine learning, Feasible, expansible	A single weak classifier is used
Zhao et al., (2021)	Anomaly- based	Detecting bot activity using both command and control and attack phases through Novelty Detection System	Difficult to realize full implementation of such a system on a large internal network
Samiksha *et al*., (2021)	DT, Naive Bayes, and Neural Network	stated that in the Internet's explosive expansion is a creating new avenue for network attack including fraud and data theft	The researchers did not take note of cybercrime when expose to a larger network attacks
Shah, Sharma and Bandgar 2021	NLP, NN, CNN	Developed a customized platform and implemented anomaly post detection	The intrusion detection only detected cyber bullying attack
Arora and Gosain 2021	MariaDB, MYSQL,TPC-H	Developed IDS for data warehouse with second level authentication to reinforce access control security	Attackers used their attacking techniques to overcome the second-level authentication

Ali, et al., (2021)	RF,SVM, Decision Tree, & Naive Bayes Classifiers	Developed a model to detect External threats &Internal compromised devices.	The result shows that more than 70% of data breaches are caused by internal users
Rovito *et al*., (2022)	DT. KNN	Proposed Evolutionary Computation Approach for Bot Detection in the bank network	The model was not efficient when compared with the existing model
Jefferson and Duarte, (2022)	Digital fingerprint, Naive Bayes,	Concentrate on using intelligent models to detect the presence of botnets in network traffic	Computational overhead will affect real-time
Fernández el at., (2022)	Decision Tree, Naive Bayes Classifier	Developed Bot detection technique for detection of social medial platform	Detects Anti-viruses and not bot
Chen et al., (2022)	LR, SVM, Naive Bayes Classifier	Transformation raw data and information into meaningful and actionable knowledge for the organization	Detected Fake account and Anti-viruses
Abdel Karim Kassem, (2022)	SVM,DT,RF and KNN	Created an intelligent system for cyber intrusion detection and security assessment with machine learning techniques.	Detected attacks based on the abnormality of the Network
Hanaa and Sultan 2023	Logistic Regression and Support Vector Machine	Focused was to develop a state-of-the-art security solution based on honeypot technology	Degradation in performance when exposed to large dataset
Alsubaei, (2023)	RF classifier	Developed The Detection of Unsuitable tweeter Connected to Fake Accounts.	Detects base on the fake accounts.
Francisco et al., 2023	DT, KNN, LR, Naive Bayes and Bag of Words (BOW) model	Use Host Intrusion Detection Systems (HIDS) to discover aberrant activities in Internet of Things devices with 89.75%	Difficult to detect bots but uses other terminologies not captured by BOW in larger dataset
Berkant *et al.,* 2023	RF, NB, LF, and KNN	Proposed Network intrusion detection system by learning jointly from tabular and text-based features	There is room for improvement in precision when compared to certain previousstudies
Etuh et al. 2023	RF, SVM, DT, LR Classifier	Developed Intelligent Intrusion Detection model for Prevention of attacks on Bank network platform.	Degradation in processing speed when exposed to large dataset and during prevention
Tewogbade *et al*., (2024)	DT and RF) & unsupervised learning (Deep Learning)	developed a botnet attack detection in the Internet of Things using machine learning models	Detected attacks based on the abnormality of the Network.

Syeda *et al*., (2024)	Built a network-based method for detection and prevention that uses processes based on anomaly detection.	The researchers faces Challenges during the process of Implementation

Research Gap

The major weaknesses identified in the existing systems includes: All the literature reviewed that focused on intrusion detection models did not address the unique detection of intrusions in bank networks. All the approaches used in the existing studies are only reactive and not proactive. The activation of the recommended protection features may come long after the assault has taken place, potentially devastating the targeted user. However, the proposed study intelligently and proactively detect cybercrimes attacks with the use of machine learning algorithm to proactively identify fault attacks in the financial network system from launching attacks on the user that has not been developed. This forms the research gap for this work

Research Methodology

The research work adopted the experimental research methodologies. Observations and literature review helps in the exploration of concepts and models of the existing system.. The Unified Modelling Language (UML), Draw.io, and Visual Studio Code were used to design a conceptual architecture for an Enhanced Classifiers. Rapid Application Development (RAD) methodology, python programming language, and MySQL were used to develop an enhanced cybercrime detection model and evaluate the performance of the new system and the existing system in terms of accuracy, precision, detection rate, F1 Score, and Receiver Operating Characteristics (ROC) Curve using a generated dataset of size of over two million.

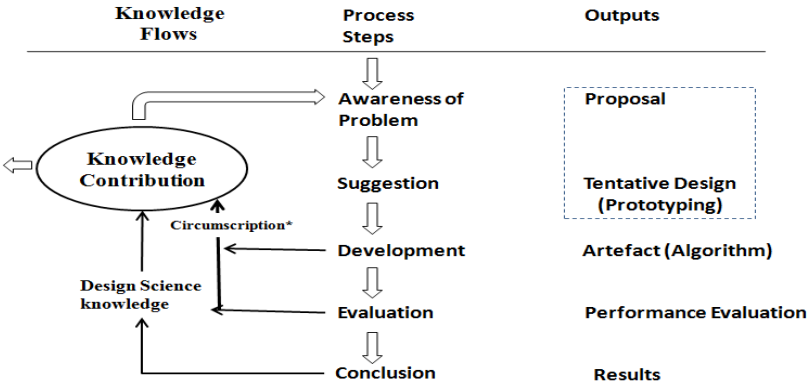

Figure 2: Experimental design science method

Figure 2 illustrates the Design Science Research (DSR) Process Model, a framework commonly used in Information Systems and Computer Science to develop and evaluate artifacts such as algorithms and prototypes that address specific problems. The process begins with an awareness of a problem, where researchers identify a practical issue that needs solving. This is followed by the suggestion phase, in which potential solutions are proposed. Next, during the development phase, a tentative design or prototype is created. This design is then put

through evaluation to determine how effectively it addresses the problem. Finally, the conclusion phase involves summarizing the findings and reflecting on the results of the entire process. Each step in the process produces specific outputs. The suggestion phase results in a proposal, while development yields a tentative design or prototype. The outcome of building the solution is referred to as an artifact, such as an algorithm. The evaluation step provides performance data, and the conclusion generates final results. On the left side of the diagram, the knowledge flows show how existing design science knowledge informs the process. Through a process called circumscription, the knowledge is refined iteratively based on feedback and evaluation results. This iterative process ensures continuous improvement and leads to knowledge contribution, which is the central goal of DSR. The knowledge generated is then fed back into the scientific community, enriching the broader body of design science knowledge.

The figure 2 also highlights important feedback loops. These loops, especially between development, evaluation, and suggestion, indicate that the process is not linear but iterative. Designs may be revised and refined multiple times based on evaluation outcomes. Enhanced models are iterative and emphasize both the creation of practical solutions and the generation of theoretical knowledge. It demonstrates a continuous and reciprocal relationship between real-world problem-solving and scientific knowledge advancement.

Results and Discusion

A Voting Classifier improves predictive performance by pooling the outputs of several diverse base models such as Decision Trees, Logistic Regression, and K-Nearest Neighbors into a single consensus prediction. It can employ hard voting, where each model casts a categorical vote and the majority class wins, or soft voting, where the models' predicted probabilities are averaged and the class with the highest mean probability is chosen. Because its members are deliberately varied and tend to make different, uncorrelated errors, their aggregation lowers variance and enhances generalization; as a result, the combined model is usually more stable and reliable, particularly on noisy or imbalanced data. In the present comparison this ensemble achieved precision, recall, and accuracy almost indistinguishable from those of the strongest standalone learner (the Random Forest), underscoring how complementary strengths across models translated into near-top-tier performance.

By contrast, a Stacking Classifier (or stacked generalization) goes beyond simple aggregation: it first trains multiple base learners, then feeds their individual predictions into a second-level, meta-model often a Logistic Regression or a Decision Tree that learns how best to blend those outputs. This meta-model can capture nuanced relationships and dependencies among the base predictions, effectively discovering which models are most trustworthy for which instances. Consequently, stacking is capable of uncovering complex data patterns that individual models or straightforward voting schemes might overlook, yielding superior predictive power. In the current evaluation the Stacking Classifier posted the highest scores across every metric, including AUC, demonstrating its ability to integrate the strengths of each base learner while muting their weaknesses.

Voting Classifier chiefly enhances stability by averaging or majority-selecting among diverse but comparably strong models, whereas a Stacking Classifier attains greater adaptability and often the best overall accuracy by training a meta-model to weight base predictions intelligently.

Model	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1-Score (%)**	**AUC (%)**
Random Forest	99.97	99.98	99.97	99.98	99.18
Decision Tree	99.77	99.50	99.60	99.50	99.60
Logistic Regression	99.33	99.10	99.20	99.10	99.10
KNN	99.50	99.30	99.30	99.30	94.02
Voting Classifier	**99.97**	**99.98**	**99.97**	**99.97**	**98.30**
Stacking Classifier	**99.98**	**99.98**	**99.98**	**99.98**	**99.22**

Table 1: The comparison between Existing classifiers and Enhanced models on CIC-IDS2017 datasets

The six classification models based on five evaluation metrics: Accuracy, Precision, Recall, F1-Score, and AUC (Area Under the ROC Curve)**.** The models evaluated include four individual classifiers (Random Forest, Decision Tree, Logistic Regression, KNN) and two ensemble methods (Voting Classifier and Stacking Classifier). The table compares six classification models Random Forest, Decision Tree, Logistic Regression, K-Nearest Neighbors, Voting Classifier, and Stacking Classifier using five evaluation metrics: accuracy, precision, recall, F1-score, and area under the ROC curve (AUC). Accuracy measures the proportion of correct predictions, precision reflects the share of true positives among all positive predictions, recall represents the share of true positives detected by the model, F1-score balances precision and recall, and AUC gauges each model's ability to separate classes across thresholds.

Among the individual algorithms, Random Forest stands out with near-perfect accuracy (99.97 %), precision (99.98 %), recall (99.97 %), F1-score (99.98 %) and a high AUC of 99.18 %, demonstrating both consistency and robustness. The single-tree Decision Tree model follows closely, achieving 99.77 % accuracy and a 99.60 % AUC, though it is slightly more vulnerable to overfitting. Logistic Regression, despite being a simple linear model, reaches respectable values 99.33 % accuracy and 99.10 % AUC but lags behind the tree-based approaches and ensembles. KNN attains solid accuracy (99.50 %) yet its AUC drops to 94.02 %, revealing less reliable class-separation, likely due to sensitivity to data distribution.

Turning to ensembles, the Voting Classifier which aggregates predictions by majority vote matches Random Forest's accuracy (99.97 %) and nearly its precision and recall, though its AUC (98.30 %) is modestly lower. The Stacking Classifier, which feeds multiple base-model outputs into a meta-learner, edges out all rivals: it posts 99.98 % across accuracy, precision, recall, and F1-score, alongside a leading AUC of 99.22 %. The Stacking Classifier emerges as the overall best performer owing to its balanced, top-tier metrics; Random Forest and the Voting ensemble are strong alternatives with only slightly reduced AUC; Decision Tree provides solid, interpretable results; whereas Logistic Regression and KNN are satisfactory but less effective, the latter notably weaker in AUC.

Table 2: The comparison between |Existing studies and an Enhanced models on CIC-IDS2017 datasets

Authors	Accuracy	Precision	Recall	F1 Score	ROC
Enhanced Classifier	**99.98%**	**99.98%**	**99.99%**	**99.99%**	**99.90%**
Berkant *et al*,.2024	99.90%	99,40%	99.90%	99.60%	89.90%
(Ho *et al.*, 2022)	98.5%	98.2%	99.0%	98.6%	87.7%
(Hassan et al. 2021)	96.2%	95.5%	95.9%	96.1%	84.5%

The Enhanced Classifier achieves nearly perfect results, with an accuracy of 99.98%, meaning it correctly predicts almost every case. Its precision is also 99.98%, showing it rarely mislabels negative cases as positive. The recall of 99.99% indicates it detects almost all true positive cases. With an F1 score of 99.99%, it balances precision and recall exceptionally well. Its ROC score of 99.90% demonstrates excellent ability to distinguish between positive and negative cases. Overall, this classifier performs outstandingly across all metrics.

The model by Berkant et al., 2024 performs very well with 99.90% accuracy, making very few errors. Its precision is slightly lower at 99.40%, so it produces a few more false positives than the enhanced classifier. It has a high recall of 99.90%, detecting nearly all positive cases. The F1 score of 99.60% reflects a strong balance between precision and recall. However, its ROC score of 89.90% is noticeably lower, suggesting it is less effective at differentiating between classes in some cases.

Ho et al., 2022 shows solid performance with an accuracy of 98.5%, correctly predicting most cases. The precision is 98.2%, indicating more false positives compared to the newer models. Recall at 99.0% shows it detects most positive cases. The F1 score of 98.6% highlights a good balance but is lower than more recent classifiers. Its ROC score of 87.7% points to a weaker ability to distinguish between classes. The model by Hassan et al., 2021 is less accurate, at 96.2%, with more incorrect predictions. Its precision of 95.5% means it has a higher false positive rate. Recall is 95.9%, so it misses more true positives compared to others. The F1 score of 96.1% is decent but not as strong. Its ROC score of 84.5% is the lowest, indicating the least capability to separate positive and negative cases.

In summary, there is a clear improvement from Hassan et al. (2021) through Ho et al. (2022) and Berkant et al. (2024), culminating in the Enhanced Classifier which performs almost flawlessly across all evaluation metrics. This progression reflects advancements in correctly identifying cases, minimizing false results, and effectively distinguishing between classes.

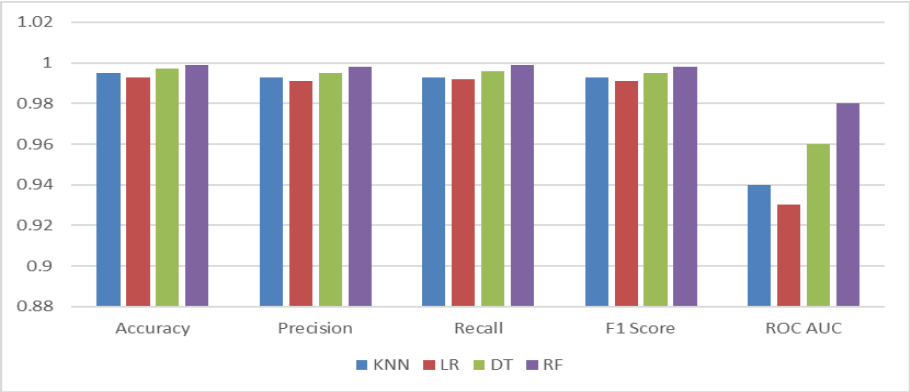

Figure 3: Performance of each Classifiers

Figure 3 presents the bar chart compares four machine learning classifiers KNN, Logistic Regression (LR), Decision Tree (DT), and Random Forest (RF) across five key evaluation metrics: Accuracy, Precision, Recall, F1 Score, and ROC AUC. Accuracy measures the proportion of correct predictions out of all predictions made. Precision reflects how many of the positive predictions were actually correct. Recall shows how many of the actual positive cases were correctly identified. The F1 Score balances precision and recall by calculating their harmonic mean. ROC AUC evaluates how well the model distinguishes between positive and negative classes, with higher values indicating better discrimination. From the chart, Random Forest consistently scores highest across all five metrics, demonstrating superior performance. Decision Tree performs well too, especially in ROC AUC, where it ranks just below Random Forest. KNN and Logistic Regression have slightly lower scores, with a noticeable drop in ROC AUC compared to Decision Tree and Random Forest. This suggests they are less effective at distinguishing between classes. Despite differences, all models show high Accuracy, Precision, Recall, and F1 Scores, indicating they generally classify well. The most significant variation among models is in ROC AUC, highlighting the advantage of Random Forest and Decision Tree in class separation.

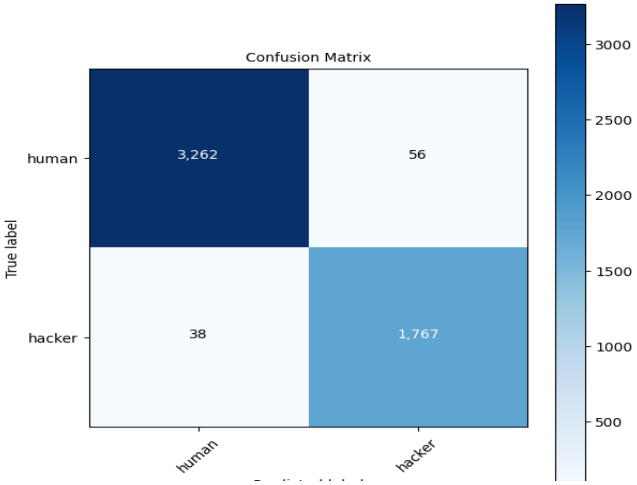

Figure 4: Matrix for Enhanced Model

Figure 4 presents matrix for voting and stacking classifier, that Out of the 5 123 network sessions evaluated, the classifier correctly labelled 3 262 of the 3 318 genuine-human sessions and 1 767 of the 1 805 genuine-hacker sessions. Only 56 humans were mistakenly flagged as hackers (a 1.7 % false-positive rate), while 38 cybercrimes slipped through as humans (a 2.1 % false-negative rate). Overall accuracy: 98.2 % (5 029 / 5 123). Precision for the "hacker" class: 96.9 % (the model's hacker alarms are correct nearly 97 times out of 100). Recall for the "hacker" class: 97.9 % (it detects almost 98 % of all hackers present). F1-score for the "hacker" class: 97.4 % (harmonising precision and recall). Visually, the confusion-matrix heatmap places almost every count on its main diagonal, confirming that the classifier distinguishes the two classes with high reliability. The slightly higher number of human→ cybercrime errors (56) than cybercrimrs→human errors (38) indicates a marginal bias toward caution—preferring to err on the side of flagging suspicious traffic rather than letting potential intrusions pass unnoticed. Overall, the model demonstrates excellent discrimination between normal human activity and malicious behaviour.

Summary, Conclusion and Recommendation

a). Summary

This study proposed an enhanced ensemble classifier for the detection of cybercrimes, particularly botnet attacks, within financial networks. The system integrates multiple machine learning algorithms into a unified architecture, leveraging ensemble strategies—specifically Voting and Stacking Classifiers—to optimize detection accuracy and resilience against sophisticated threats.

Six models were evaluated using five performance metrics: Accuracy, Precision, Recall, F1-Score, and AUC. Among the individual classifiers, Random Forest outperformed others with near-perfect scores. However, the ensemble methods, particularly the Stacking Classifier, demonstrated superior overall performance, achieving 99.98% across almost all metrics and a leading AUC of 99.22%. The experimental design followed the Design Science Research (DSR) model, progressing through awareness of the problem, solution suggestion, prototype development, evaluation, and conclusion. The proposed system was tested using the CIC-IDS2017 dataset, which simulated real-world network traffic and botnet behaviors. The analysis revealed that existing systems were largely reactive and insufficient in addressing the evolving nature of cyberattacks on financial systems. The proposed model filled this gap by offering a proactive, intelligent detection mechanism with high accuracy and generalization capabilities.

b). Conclusion

The research successfully developed and evaluated an intelligent cybercrime detection model that outperforms traditional machine learning approaches in identifying botnet-related threats in financial networks. The use of ensemble learning techniques, particularly the Stacking Classifier, allowed the model to learn from multiple algorithms and make optimized predictions, thus achieving a high detection rate while minimizing false positives. The findings confirm that ensemble methods are more effective in cybersecurity tasks due to their ability to combine the strengths of diverse base models and adapt to complex data patterns. This study also highlights the limitations of earlier models which often focused on specific aspects such as

fake accounts or malware but lacked a comprehensive and proactive detection capability for botnet-based financial crimes.

c). Recommendations

i. **Adopt Ensemble-Based Intrusion Detection Systems (IDS)**: Financial institutions should integrate stacking and voting classifiers into their security architecture to enhance the detection of advanced cyber threats, particularly botnet intrusions.

ii. **Extend Dataset Coverage and Real-Time Deployment**: Future work should explore more extensive real-time datasets to test the model's effectiveness in live environments and further improve its generalization to unknown threats.

iii. **Continuous Model Updating**: Given the evolving nature of cyber threats, the detection model should be periodically retrained on fresh data to ensure its relevance and accuracy in identifying emerging attack vectors.

iv. **Policy and Awareness Campaigns**: Alongside technical solutions, institutions must improve internal policies and educate staff and users to reduce human vulnerabilities that attackers often exploit.

References

Abdullah, S., Kopyrski, P., Roaf, J., Shabunina, A., van Elkan, R. and Xu, X. C. (2020). "Twitter Bot Detection Using Supervised Machine Learning." *Journal of Physics*: *Conference Series* 1950(2021):1-11.

Abdulrahman, A. A., and Ibrahim, M. K. (2018). Evaluation of DDoS attacks Detection in a New Intrusion Dataset Based on Classification Algorithms. *Iraqi Journal of Information & Communications Technology*. 1(3): 49-55.

Ahmad, T., and Aziz, M.N. (2019). Data Preprocessing and Feature Selection for Machine Learning Intrusion Detection Systems. *ICIC Express Letter*. 13(2): 93-101.

Arora, A., Ashanyata S., and Anjana, G. (2021). "Intrusion Detection System for Data Warehouse with Second Level Authentication." *International Journal of Information Technology*:77-89.

Aydin, M.A., Halim, Z.A., and Gokhan, C. K. (2019). "A hybrid intrusion detection system design for computer network security" *Computers and Electrical Engineering* 35(3).

Baba, C., Batog, C., Flores, E., Gracia, B., Karpowicz, I., Kopyrski, P., Roaf, J., Shabunina, A., van Elkan, R. and Xu, X. C. (2020). Fintech in Europe: Promises and Threats. *IMF* Working Paper.12-20.

Bacher H., Partam, E.C. and Karthikeyan, E. S. (2015). A feature selection algorithm using correlation based method. *Journal of Algorithms and Computational Technology*, 6(3): 385-394.

Bakhoum, T., and Ezzat, G. (2011). "Intrusion detection model based on selective packet sampling." EURASIP *Journal on Information Security* 1(2011): 1-12.

Barone, A. (2022). What Is a Bank? *http*://www.investopedia.com/terms/b/bank.asp.

Berkant D. Aykut Ç. Uğur U. and Hasan D. (2024) Network intrusion detection system by learning jointly fromtabular and text-based features, *International Journal of Information Technology*: 34-83

Blockchain for next generation services in banking and finance: cost, benefit, risk and opportunity analysis. *Journal of Enterprise Information Managemen* 34(3), 884-899.

Buba, M. N., Khwaja, A. A., Nadeem, A., Ahmad, H. F., Khan, M. K., and Blauner, C. (2020). Cybersecurity and the Banking Sector. *Hampton Roads International Security Quarterly*, 8(2) 54-60.

Calleja, S. V., Emanuele, P., Gavin, H., William, B., Yosef, H., Fayez, J., Hyder, A., and Segu, M. (2021). "Twitter Bot Detection Using Supervised Machine Learning." *Journal of Physics*: Conference Series 1950(2021):1-11.

Calvin, N., Anthony, V., and Lynne, A. (2021). Banking Cybersecurity Culture Influences On Phishing Susceptibility, *Computer Science and Information Technology* (CS & IT) http/doi: 10.5121/csit.2021.112405. 2-80

Daniel, E. (2016). "The Usefulness of Qualitative and Quantitative Approaches and Methods in Researching Problem-Solving Ability in Science Education Curriculum." *Journal of Education and Practice* (15) 1-100. http/doi: 2222-288X.

Das, S. R. (2019). The future of fintech. *Financial Management, IEEE Transactions on Network and Service Management* 8(4), 981-1007.

Das, S., Saha, S., Priyoti, A. T., Roy, E. K., Sheldon, F. T., Haque, A., & Shiva, S. (2021). Network intrusion detection and comparative analysis using ensemble machine learning and feature selection. *IEEE Transactions on Network and Service Management*, *19*, 4821–4833.

Delplace, A., Hermoso, S., and Anandita, K., (2020). Cyber Attack Detection thanks to Machine Learning Algorithms. http/arXiv preprint arXiv.06309.

Dimitris, V. (2011). Botnet lab creation with open source tools and usefulness: *Rochester Institute of Technology* RIT Digital Institutional Repository, https:/ /repository .rit.edu/theses, repository@rit.edu.

Dollah, R., Fadhlee M. (2018). "Machine learning for HTTP botnet detection using classifier algorithms." *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)* 10.1-7: 27-30.

Etuh, E. Bakpo, F. S., and Eneh, H. A. (2023). "Social Media Network Attacks and Their Preventive Mechanisms: Thesis" edited by D. C. Wyld and D. Nagamalai. Computer Science and Information Technology 9-12.

Etuh, E., Francis, S. B., and Eneh, A. H. (2021). "Social Media Network Attacks and Their Preventive Mechanisms: A Review," *Computer Science and Information Technology* (CS & IT) http/doi: 10.5121/csit.2021.112405. 10(8): 12-40

Francisco, L.C.F., Samuel, C.M.S., Rafael, Z.A.M., Fábio, L.L.M., and Rafael, T., (2023). Botnet Detection and Mitigation Model for IoT Networks Using Federated Learning, *Electrical Engineering Department* (ENE), *Technology*

Hanaa, A., and Sultan, A. (2023). Developing a Security Scheme for Banking Networks Based On Honeypot Technology. *International Journal of Engineering Research and Technology*, 12-25.

Ho, C. M. K., Yow, K.-C., Zhu, Z., & Aravamuthan, S. (2022). Network intrusion detection via flow-to-image conversion and vision transformer classification. *IEEE Access*, *10*, 97780–97793.

Ioulianou B.M, and Philokypros, T., (2018) "A signature-based intrusion detection system for the Internet of Things." *Information and Communication Technology*. (3-12)

Jefferson, I., and Duarte, S. (2022). Botnets and how to automatic detect them: exploring new ways of dealing with botnet classification, *Institution*:, Address: RN 118, S/N, Povoado Base Física, Zona Rural, Ipanguaçu - RN, *CEP*: 59508-59510,

Jing, W., and Ioannis, C., (2016). Botnet Detection based on Anomaly and Community Detection, http://dx.doi.org/10.1109/*TCNS*.2532804

Jithu, P., Jishma, S., Aiswarya, R., Haripriya, A. P. (2021). Intrusion Detection System for IOT Botnet Attacks Using Deep Learning, *Singapore* Pte Ltd.

Lallie, H. S., Shepherd, L. A., Nurse, J. R.C., Erola, A., Epiphaniou, G., Maple, C. M. A., Song, H., Alshamari, M. and Cao, Y. (2021). A Survey on Blockchain Technology: Evolution, Architecture and Security. *IEEE Access*, 9, 61048- 61073.

Li, S. (2018). DDoS Attacks Detection using Machine Learning Algorithms. In International Forum on Digital TV and Wireless Multimedia Communications. *Springer*. 45

Syeda, L., Li D., Hassan J., Ayman A., Salmah F., Ahmed K. A., and Azeem K., (2024). "Ensemble learning." *Encyclopedia of biometrics*: 270-273.