Enhancing financial cybersecurity via advanced machine learning: analysis, comparison

Grace Odette Boussi[1], Himanshu Gupta[2], Syed Akhter Hossain[3]
[1]Department of Information Technology, Amity University, Noida, India
[2]Department of Information Technology, Faculty of Cyber Security, Amity University, Noida, India
[3]Department of Computer Science and Engineering, University of Liberal Arts, Dhaka, Bangladesh

Article Info

Article history:

Received Apr 22, 2024
Revised Nov 14, 2024
Accepted Nov 24, 2024

Keywords:

Cybersecurity
Deep learning
Extreme gradient boosting
Machine learning
Malware

ABSTRACT

The financial sector is a prime target for cyber-attacks due to the sensitive nature of the data it handles. As the frequency and sophistication of cyber threats continue to rise, implementing effective security measures becomes paramount. In this paper we provide a comprehensive comparison of six prominent machine learning techniques utilized in the financial industry for cyber-attack prevention. The study aims to identify the best-performing model and subsequently compares its performance with a proposed model tailored to the specific challenges faced by financial institutions. This paper looks at using advanced machine learning methods to make cybersecurity stronger for financial institutions. The work explores the deployment of cutting-edge machine learning algorithms - logistic regression, random forest, support vector machines (SVM), K-nearest neighbour (KNN), naïve Bayes, extreme gradient boosting (XGBoost), and deep learning technique (Dense Layer) - to fortify the cybersecurity framework within financial institutions. Through a meticulous analysis and comparative study, we explore the efficacy, scalability, and practical implementation aspects of various machine learning algorithms tailored to address cybersecurity concerns. Additionally, we propose a framework for integrating the most effective machine learning models into existing cybersecurity infrastructure, offering insights into bolstering resilience against evolving cyber threats. In our comparison, XGBoost exhibited outstanding performance with an accuracy of 95%.

Corresponding Author:

Grace Odette Boussi
Department of Information Technology, Amity University
Noida sector 143, 201301, Uttar Pradesh, India
Email: graceboussi@gmail.com

1. INTRODUCTION

The digital landscape has made significant advancements, especially online, where a majority of our activities take place, due to the creative methods employed by attackers, the risk of cyberattacks is rapidly increasing [1]. Rapid technological evolution and increasing internet users, reaching 4.4 billion in 2019, are expected to rise post-COVID-19. With online services holding sensitive data, attackers increasingly target hacking such platforms [2]. In today's digital era, the financial sector operates within an intricate web of interconnected systems and processes, making it a prime target for cyber threats of unprecedented sophistication and scale [3]. As digital transactions, sensitive financial data, and complex networks become increasingly common, traditional cybersecurity measures often prove insufficient in protecting against evolving threats. Consequently, financial institutions are under growing pressure to strengthen their defenses and mitigate the risks posed by cyber-attacks. In response to this urgent need, there is a rising interest in harnessing advanced machine learning techniques to enhance cybersecurity within the financial sector. These technologies play a crucial role in the implementation of cyber defense strategies such as monitoring, control, threat detection, and alarm systems [4]. The adoption of machine learning in cybersecurity has witnessed significant growth in popularity [5]. The current state of financial cybersecurity underscores the essential role of advanced machine learning techniques in improving defense mechanisms against cyber threats. Researchers are directing their efforts towards conducting comprehensive analyses, comparative studies, and developing integration frameworks to equip financial institutions with resilient and adaptable tools. These advancements empower the financial sector to more effectively manage cyber risks, thereby safeguarding the security and integrity of sensitive financial data and transactions.

Technology has revolutionized our lives, bringing immense convenience but also introducing a host of challenges [6]. One notable issue is the escalation of cybersecurity threats due to the rapid advancement of technology. Another concern is the exponential growth of data volumes , making it increasingly challenging to ensure security. Moreover, highly skilled hackers with extensive knowledge of systems and programming have the ability to exploit well-protected systems, compounding security concerns [7]. The term "malware" is a fusion of "malicious code" and "malicious software," denoting software designed with the primary aim of gaining unauthorized access to external tools. Furthermore, malware has the potential to inflict enduring damage on both individuals and organizations [8]. The increasing release of malware is worrying security experts worldwide. It's important for researchers and the security community to stay updated on new types of malware and how to detect them [9]. Cybersecurity is increasingly emphasizing the identification and suppression of malware [10]. This shift reflects the growing recognition of the significant threat posed by malicious software to computer systems, networks, and data. As cyber threats continue to evolve and become more sophisticated, detecting and mitigating malware has become a top priority for cybersecurity professionals and organizations [11]. By implementing robust detection and mitigation strategies, cybersecurity experts aim to safeguard digital assets, prevent unauthorized access, and minimize the impact of malware-related incidents on individuals, businesses, and critical infrastructure. Researchers are interested in using machine learning and deep learning because they can create advanced models for detecting complicated malware [12].

This research endeavours to delve into the realm of sophisticated machine learning methodologies and their application in enhancing cybersecurity within financial institutions. Through a comprehensive analysis and implementation comparison, this study seeks to elucidate the efficacy, scalability, and practical implications of various machine learning algorithms tailored specifically for financial cybersecurity. By examining the strengths and limitations of different approaches, we aim to provide insights into the optimal utilization of machine learning techniques to address the unique challenges faced by financial institutions in safeguarding their digital assets and infrastructure.

The overarching objective of this research is twofold. Firstly, to assess the performance and suitability of advanced machine learning algorithms in detecting and mitigating cyber threats within the financial sector. Secondly, to propose a framework for the integration and implementation of these techniques into existing cybersecurity infrastructure. By undertaking this endeavour, we endeavour to contribute to the advancement of cybersecurity practices in financial institutions, paving the way for more resilient and adaptive defence mechanisms in the face of evolving cyber threats.

2. LITERATURE REVIEW

Malicious software, commonly referred to as malware, can severely degrade device performance and pose a risk of data misuse by attackers once a device is affected. Moreover, evolving malware types make conventional detection techniques cumbersome and ineffective for identifying new and generic variants [13]. Implementing machine learning and deep learning method in order to reduce the impact of cybercrime has been a remarkable work which has been done by many authors. Roponena *et al.* [14] said that machine learning plays a critical role in cybersecurity solutions by enabling the automatic analysis of data patterns and learning from them to prevent similar attacks or forecast potential threats. Currently, machine learning methods assist cybersecurity professionals in rapidly identifying various types and attributes of malware [15], [16]. Bokolo *et al.* [17] compares seven machine learning and deep learning methods to detect malware using byte, opcode, and section codes. The study aims to accurately classify malware into nine distinct families by extracting and merging byte, section, and opcode data. Techniques include random forest, decision tree, support vector machines (SVM), K-nearest neighbour (KNN), stochastic gradient descent (SGD), logistic regression, naïve Bayes, and deep learning. On their side, Ouahab *et al.* [18] introduce a novel method for identifying unknown malware types using machine learning and visualization. Three efficient classifiers achieve up to 98% precision in malware classification. For [19], a method combining

SVM classifiers and active learning by learning (ALBL) addresses limited labelled data in malware classification was proposed, they evaluation it using the Microsoft Malware classification challenge dataset on Kaggle and ALBL demonstrated the capability to enhance model performance.

Numerous studies have explored the development of effective malware classifiers, with [20] showcasing the use of the KNN algorithm. Additionally, researchers have delved into utilizing deep learning networks to enhance malware classification performance. Convolutional neural networks, as demonstrated in [21], [22], and recurrent neural networks have been employed to identify both traditional and concealed malware [23]. Despite these advancements, identifying entirely new malware variants remains challenging. Ouahab *et al*. [24] introduced a method for detecting upcoming malware generations. This involved training random forest and KNN models on 24 distinct malware families. Venkatasubramanian *et al*. [25] worked on IoT malware analysis, they outline various methods that combine federated learning (FL) with IoT by exploring the practical uses of FL, research obstacles, and future research paths. Halbouni *et al*. [26] examined intrusion detection systems, focusing on machine learning and deep learning algorithms combatting malicious behaviour, they explored recent advancements in network implementations, algorithms, and datasets for effective detection systems. In their study, Jin *et al*. [27] introduced a malware detection method employing deep learning, utilizing an autoencoder to discern malware's functional traits. Achieved accuracy stands at 93% [27].

Sethi *et al*. [28] devised a malware detection framework utilizing the Cuckoo sandbox for dynamic file analysis, integrating Chi square and random forest techniques for feature selection, with decision tree classifiers achieving the highest accuracy. Darem *et al*. [29] introduced a model leveraging concept drift detection and sequential deep learning, achieving 99.41% accuracy for new malware variants. Wu *et al*. [30] addressed unbalanced datasets using a three-tier cascading extreme gradient boosting (XGBoost) approach and cost-sensitive learning techniques, demonstrating the effectiveness of XGBoost in malware detection. McGiff *et al*. [31] enhanced malware detection by combining hardware features and permission data, resulting in improved model performance. Anuar *et al*. [32] proposed opcode analysis for malware detection, showing higher occurrence frequencies in malware compared to benign applications, suggesting its significance in malware classification.

Some survey were also conducted and this is the case with [33] who offered a comprehensive review of recent cybersecurity works employing deep learning in mobile and wireless networks, encompassing infrastructure threats, software attacks, and privacy protection. They presented detailed deep learning techniques, examined cybersecurity works, discussed challenges, implementation details, and solution performance, identifying the most effective deep learning methods for various threats and attacks. In their study, the authors [34] leveraged the machine learning malware detector (MLMD) program to automate static and dynamic analysis processes. They trained XGBoost models on datasets from both analyses, achieving detection accuracies of 91.9% and 98.2%, along with sensitivities of 96.4% and 98.5% for static and dynamic datasets, respectively.

To improve cybersecurity, significant efforts have been made in the last decade to utilize machine learning approaches effectively. Enhancing security in the complex technical landscape requires a cautious and strategic approach to address the evolving cyber threats [35] and Malware variants continue to evolve through the use of advanced packing and obfuscation techniques, posing increased challenges for their classification and detection [36]. As the internet expands and social media becomes more widespread, data breaches have consequently become a primary concern in the realm of cybersecurity [37].

3. METHOD

The methodology encompasses data collection, model selection, experimental design, and performance evaluation. We utilize diverse datasets reflecting varied cyber-attack patterns and normal operational data. Selected models, including logistic regression, random forest, SVM, KNN, naïve Bayes, XGBoost, and deep learning, are chosen for their effectiveness in anomaly detection. Pre-processing involves data cleaning, normalization, and feature engineering. Hyperparameter tuning optimizes model performance. Evaluation metrics such as accuracy, precision, recall, F1-score, and area under the curve (AUC) assess model effectiveness. Data sourced from the Canadian Institute for Cybersecurity consists of 11,598 rows and 471 columns, with five labels representing different classes.

3.1. Process outline

The research process is shown in Figure 1. The diagram provides an overview of the workflow, and each step is explained in more detail in the text that follows the diagram. Every stage of the process is clearly outlined to help make the information easier to understand and follow.

The process described outlines a comprehensive workflow for developing and evaluating machine learning and deep learning models. Let's break it down step by step:

Step 1: Data pre-processing and training for machine learning models:

- The process begins with preparing the data for training machine learning models. This involves steps like cleaning the data, handling missing values, encoding categorical variables, and scaling features.
- Once the data is prepared, it's split into training and testing sets. The training set is used to train the machine learning models, while the testing set is reserved for evaluating their performance.

Step 2: Selection of machine learning models based on analysis of variance (ANOVA):

- ANOVA is a statistical method used to compare the means of different groups to determine if there are significant differences between them.
- In this step, ANOVA is employed to compare the performance of various machine learning models on the training data. This helps in selecting the most promising machine learning algorithms for further evaluation.

Step 3: Training and testing five different machine learning models:

- After selecting the machine learning models based on ANOVA, the next step involves training and testing these models on the dataset. This allows for assessing their performance in terms of metrics such as accuracy, precision, recall, and F1-score.
- The evaluation of each model provides insights into its strengths and weaknesses, aiding in the selection of the best-performing machine learning algorithm.

Step 4: Engagement with deep learning and model customization:

- Moving beyond traditional machine learning, the workflow transitions to exploring deep learning models.
- Prior to training deep learning models, the data undergoes pre-processing similar to the machine learning phase. Once pre-processed, deep learning models are constructed and customized.
- Model customization involves adjusting the architecture, hyperparameters, and other settings to enhance performance on the given task.
- However, despite customization efforts, the performance of the deep learning model falls short of expectations.

Step 5: Transfer learning with machine learning (XGBoost):

- In response to the suboptimal performance of the customized deep learning model, a decision is made to explore alternative approaches.
- Transfer learning is employed, where features learned from the deep learning model are transferred to a traditional machine learning model, specifically XGBoost.
- XGBoost, known for its robustness and performance, is selected for its ability to handle complex datasets effectively, it employs decision tree-based techniques to classify malicious executables through a gradient boosting approach [38].

Step 6: Comparison of machine learning, deep learning, and XGBoost results:

- The final step involves comparing the performance of the machine learning models, deep learning model, and XGBoost.
- Metrics such as accuracy, precision, recall, and F1-score are used to evaluate and compare the models.
- Based on the comparison, XGBoost emerges as the top-performing model, surpassing both traditional machine learning and customized deep learning approaches.

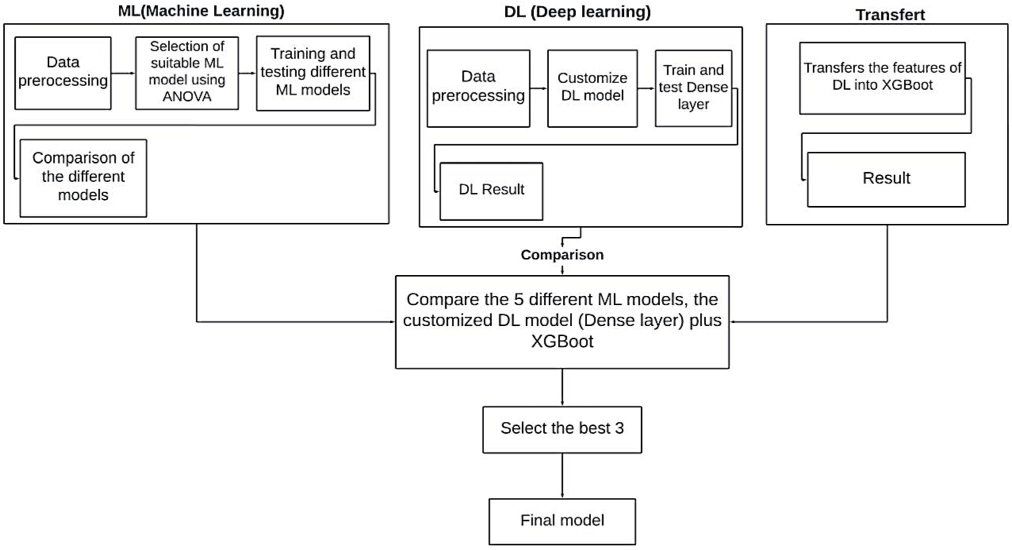
ML(Machine Learning)
Data prerocessing
Selection of suitable ML model using ANOVA
Training and testing different ML models
Comparison of the different models
DL (Deep learning)
Data prerocessing
Customize DL model
Train and test Dense layer
DL Result
Transfert
Transfers the features of DL into XGBoot
Result
Comparison
Compare the 5 different ML models, the customized DL model (Dense layer) plus XGBoot
Select the best 3
Final model

Figure 1. Process outline

3.1.1. Machine learning models

Our work is done using machine learning and deep learning, so in this part, we will be talking about the steps used in order to train our machine learning models. Here are the five machine learning which have been used: naïve Bayes; random forest; logistic regression; SVM; and K-nearest. During our training of various machine learning models, we observed that random forest exhibited strong performance, while naïve Bayes performed the least effectively. Table 1 demonstrates their respective performance.

a) Data pre-processing: in the preparation of our data for training our diverse machine learning models, we have dropped less important classes and features by running these codes:

```
X = df.drop(columns=['Class']) # Features
y = df['Class'] # Target
```

b) Split our data into training and testing where 80% of data were for training and the remaining 20% for testing.# Split the data into training and test sets

```
X_train, X_test, y_train, y_test = train_test_split(X, y, stratify=y,test_size=0.2, random_state=42)
```

c) We have used ANOVA -based to help us selecting our features:

```
num_features_to_select = 120
selector = SelectKBest(score_func=f_classif, k=num_features_to_select)
X_train_selected = selector.fit_transform(X_train, y_train)
X_test_selected = selector.transform(X_test)
```

Table 1. Machine leaning models performance comparison

S.N	Models	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
1	Logistic Regression	0.80	0.80	0.80	0.80
2	SVM	0.82	0.82	0.82	0.82
3	Random Forest	0.94	0.94	0.94	0.94
4	KNN	0.90	0.90	0.90	0.90
5	Naïve Bayes	0.58	0.69	0.58	0.54

3.1.2. Deep learning model

Following the training of our data using machine learning, we utilized the same data and class to construct our model using deep learning.

a) Data pre-processing

1- We prepared data by extracting the input features (X) and the target variable (y) for training a predictive model.

```
X = data.iloc[:, :-1].values
y = data.iloc[:, -1].values
```

b) Convert labels to start from 0

```
y -= 1
```

c) Convert target labels to one-hot encoding

```
num_classes = len(np.unique(y))
y_encoded = to_categorical(y, num_classes=num_classes)
```

d) Split the data into training and testing sets

```
X_train, X_test, y_train, y_test = train_test_split(X, y_encoded, test_size=0.2, random_state=42)
```

e) Normalize the features

```
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)
```

f) Then we build our model

```
model = Sequential()
model.add(Dense(128, input_dim=X_train.shape[1], activation='relu'))
model.add(Dense(64, activation='relu'))
model.add(Dense(num_classes, activation='softmax'))
```

Given the suboptimal performance of our current model, we have opted to transfer its features to an XGBoost machine learning model. This strategic decision is aimed at exploring the potential for achieving improved results compared to the performance of our previously trained models.

3.1.3. XGBoost

After our model's underperformance, we transferred its features to XGBoost, where it demonstrated significant improvement.

```
y = data['Class']
y = y - 1
X_train, X_test, y_train, y_test = train_test_split(X_scaled, y, test_size=0.2, random_state=42)
# Initialize models
xgb_model = XGBClassifier(random_state=42)
# Train and evaluate each model
xgb_accuracy,xgb_precision,xgb_recall,xgb_f1 = train_evaluate_model(xgb_model, X_train, y_train, X_test, y_test)
```

4. RESULTS AND DISCUSSION

After extensive testing, XGBoost, random forest, and Dense layer emerged as the top models for malware prevention. XGBoost demonstrated exceptional performance with a 95% accuracy rate, earning its selection as the final deployed model. Evaluation metrics are presented in Figure 2.

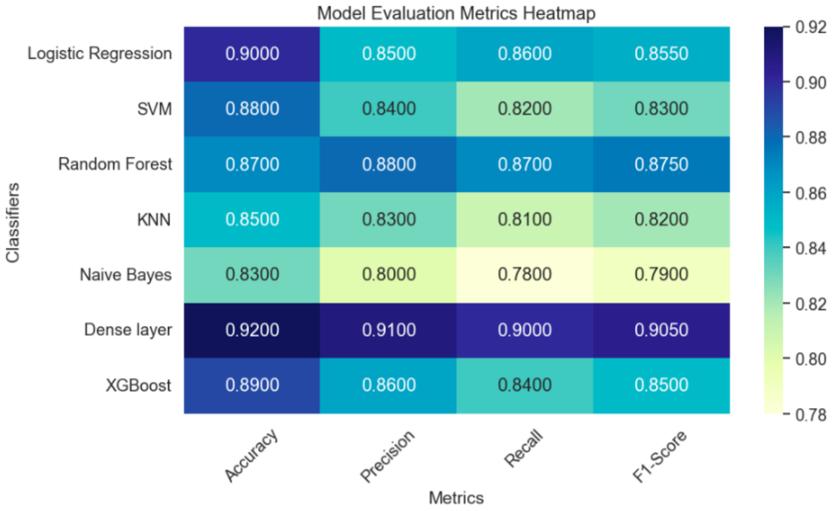

Figure 2. Model evaluation metrics heatmap

This capability is particularly valuable in tasks like malware analysis, where outliers are significant anomalies and removing them could lead to misleading conclusions. Initially, we trained our data using five machine learning models, among which random forest demonstrated strong performance with a 94% accuracy rate. Random forest, a widely used supervised machine learning algorithm, employs decision trees on multiple samples. For classification, it considers the majority vote, while for regression, it uses the average vote [39]. Since our customized model fell short at 91%, to capitalize on the strengths of our customized model, we transferred its features to XGBoost, resulting in a superior performance of 95%, surpassing even random forest. XGBoost, a high-performing machine learning algorithm, achieves exceptional accuracy by employing XGBoost. It outpaces other implementations in speed and performance, pushing computing tools for boosted tree algorithms to their limits [40]. Consequently, XGBoost was selected as our final model for its outstanding performance and ability to effectively handle dataset complexities. To highlight the significance of our results, Table 2 presents a comparative analysis of our work against existing methods. This comparison clearly demonstrates that our approach yields superior outcomes compared to the current state-of-the-art techniques.

Table 2. Comparison with the existing models

Reference	Precision (%)	Recall (%)	F-score (%)	Accuracy (%)
[41]	95	93	94	91
[42]	94	94	93	95
[43]	91	94	94	94
[44]	94	94	94	95
Our proposed model	95	95	95	95

5. COMPARATIVE ANALYSIS

The decision between traditional machine learning and deep learning models depends on factors such as the nature of the problem, data complexity, and the need for feature engineering. Each model has its own strengths tailored to different scenarios. In our case, among the machine learning models, XGBoost emerges as the standout performer with near-perfect metrics (around 0.95), showcasing its exceptional ability to handle nuances within the dataset. The performance of all models is depicted in Figure 3.

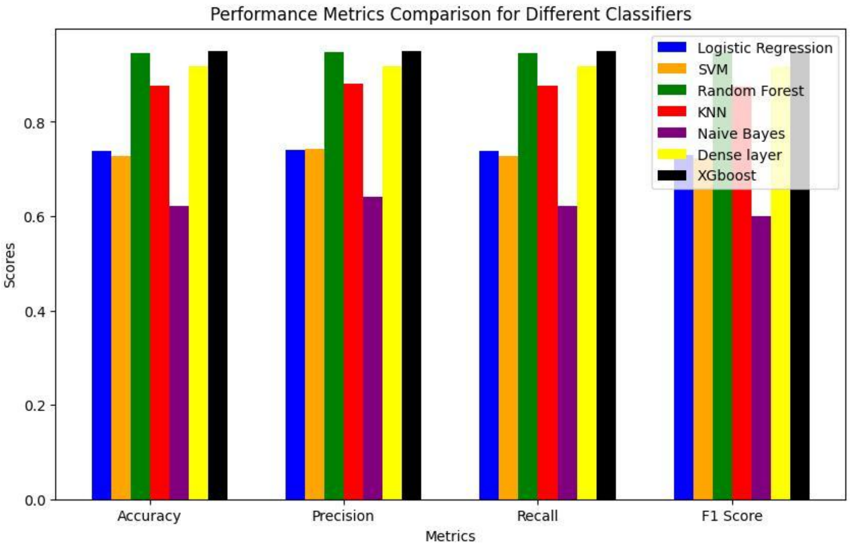

Figure 3. Performance metrics comparison for different classifiers

6. CONCLUSION FUTURE SCOPE

Our research focuses on improving cybersecurity in banks and similar institutions using advanced machine learning methods. These institutions face serious risks from cyber attacks due to the sensitive data they manage, so effective security measures are crucial. We studied how different machine learning algorithms can help detect and prevent cyber threats, especially malware. We compared six main techniques: logistic regression, random forest, SVM, KNN, naïve Bayes, and XGBoost, as well as a deep learning approach with Dense Layer. Our findings show that XGBoost performed the best, achieving an impressive accuracy of 95%. This demonstrates its effectiveness in handling complex cybersecurity data, especially for tasks like malware detection. Our study emphasizes the importance of integrating advanced machine learning models into current cybersecurity systems to better protect against evolving cyber threats. We propose a framework for implementing these techniques to advance cybersecurity practices in financial institutions. Moving forward, future research endeavours should focus on refining and optimizing machine learning models, exploring their integration into real-time threat detection systems, and expanding their application across different vectors of cyber threats. By staying ahead of cybercriminals through the strategic utilization of advanced machine learning techniques, financial institutions can fortify their cybersecurity defences and safeguard their digital assets against emerging threats.

REFERENCES

[1] M. Aljabri *et al.*, "Detecting malicious URLs using machine learning techniques: review and research directions," *IEEE Access*, vol. 10, pp. 121395–121417, 2022, doi: 10.1109/ACCESS.2022.3222307.

[2] E. Hosam, H. Hosny, W. Ashraf, and A. S. Kaseb, "SQL injection detection using machine learning techniques," in *2021 8th International Conference on Soft Computing & Machine Intelligence (ISCMI)*, 2021, pp. 15–20, doi: 10.1109/ISCMI53840.2021.9654820.

[3] S. Razaulla *et al.*, "The age of ransomware: A survey on the evolution, taxonomy, and research directions," *IEEE Access*, vol. 11, pp. 40698–40723, 2023, doi: 10.1109/ACCESS.2023.3268535.

[4] D. Dasgupta, Z. Akhtar, and S. Sen, "Machine learning in cybersecurity: a comprehensive survey," *Journal of Defense Modeling and Simulation*, vol. 19, no. 1, pp. 57–106, 2022, doi: 10.1177/1548512920951275.

[5] M. Ozkan-Okay *et al.*, "A comprehensive survey: Evaluating the efficiency of artificial intelligence and machine learning techniques on cyber security solutions," *IEEE Access*, vol. 12, pp. 12229–12256, 2024, doi: 10.1109/ACCESS.2024.3355547.

[6] Y. Li and Q. Liu, "A comprehensive review study of cyber-attacks and cyber security; emerging trends and recent developments," *Energy Reports*, vol. 7, pp. 8176–8186, 2021, doi: 10.1016/j.egyr.2021.08.126.

[7] Ö. Aslan, S. S. Aktuğ, M. Ozkan-Okay, A. A. Yilmaz, and E. Akin, "A comprehensive review of cyber security vulnerabilities, threats, attacks, and solutions," *Electronics*, vol. 12, no. 6, 2023, doi: 10.3390/electronics12061333.

[8] S. Lima *et al.*, "Artificial intelligence-based antivirus in order to detect malware preventively," *Progress in Artificial Intelligence*, vol. 10, no. 1, pp. 1–22, 2021, doi: 10.1007/s13748-020-00220-4.

[9] N. Mohapatra, B. Satapathy, B. Mohapatra, and B. K. Mohanta, "Malware detection using artificial intelligence," in *2022 13th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, 2022, pp. 1–6, doi: 10.1109/ICCCNT54827.2022.9984218.

[10] G. T. Reddy *et al.*, "An ensemble based machine learning model for diabetic retinopathy classification," in *International Conference on Emerging Trends in Information Technology and Engineering, ic-ETITE 2020*, 2020, pp. 1–6, doi: 10.1109/ic-ETITE47903.2020.235.

[11] R. Kumar and S. Geetha, "Malware classification using XGboost-Gradient boosted decision tree," *Advances in Science, Technology and Engineering Systems*, vol. 5, no. 5, pp. 536–549, 2020, doi: 10.25046/AJ050566.

[12] P. Maniriho, A. N. Mahmood, and M. J. M. Chowdhury, "A study on malicious software behaviour analysis and detection techniques: Taxonomy, current trends and challenges," *Future Generation Computer Systems*, vol. 130, pp. 1–18, 2022, doi: 10.1016/j.future.2021.11.030.

[13] U. V Nikam and V. M. Deshmuh, "Performance evaluation of machine learning classifiers in malware detection," in *2022 IEEE International Conference on Distributed Computing and Electrical Circuits and Electronics (ICDCECE)*, 2022, pp. 1–5, doi: 10.1109/ICDCECE53908.2022.9793102.

[14] E. Roponena, J. Kampars, A. Gailitis, and J. Strods, "A literature review of machine learning techniques for cybersecurity in data centers," in *2021 62nd International Scientific Conference on Information Technology and Management Science of Riga Technical University, Proceedings*, 2021, pp. 1–6, doi: 10.1109/ITMS52826.2021.9615321.

[15] J. L. G. Torres, C. A. Catania, and E. Veas, "Active learning approach to label network traffic datasets," *Journal of Information Security and Applications*, vol. 49, 2019, doi: 10.1016/j.jisa.2019.102388.

[16] K. Sethi, S. K. Chaudhary, B. K. Tripathy, and P. Bera, "A novel malware analysis framework for malware detection and classification using machine learning approach," in *Proceedings of the 19th International Conference on Distributed Computing and Networking*, 2018, pp. 1–4, doi: 10.1145/3154273.3154326.

[17] B. Bokolo, R. Jinad, and Q. Liu, "A comparison study to detect malware using deep learning and machine learning techniques," in *2023 6th International Conference on Big Data and Artificial Intelligence*, 2023, pp. 1–6, doi: 10.1109/BDAI59165.2023.10256957.

[18] I. B. A. Ouahab, L. Elaachak, Y. A. Alluhai, and M. Bouhorma, "A new approach to detect next generation of malware based on machine learning," in *2021 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies, 3ICT 2021*, 2021, pp. 230–235, doi: 10.1109/3ICT53449.2021.9581625.

[19] C. W. Chen, C. H. Su, K. W. Lee, and P. H. Bair, "Malware family classification using active learning by learning," in *International Conference on Advanced Communication Technology, ICACT*, 2020, vol. 2020, pp. 590–595, doi: 10.23919/ICACT48636.2020.9061419.

[20] I. B. A. Ouahab, M. Bouhorma, A. A. Boudhir, and L. El Aachak, "Classification of grayscale malware images using the k-nearest neighbor algorithm," in *Innovations in Smart Cities Applications Edition 3*, 2020, pp. 1038–1050, doi: 10.1007/978-3-030-37629-1_75.

[21] Y. Mourtaji, M. Bouhorma, and D. Alghazzawi, "Intelligent framework for malware detection with convolutional neural network," in *ACM International Conference Proceeding Series*, 2019, pp. 1–6, doi: 10.1145/3320326.3320333.

[22] D. Vasan, M. Alazab, S. Wassan, B. Safaei, and Q. Zheng, "Image-based malware classification using ensemble of CNN architectures (IMCEC)," *Computers and Security*, vol. 92, 2020, doi: 10.1016/j.cose.2020.101748.

[23] S. Shukla, G. Kolhe, S. M. PD, and S. Rafatirad, "RNN-based classifier to detect stealthy malware using localized features and complex symbolic sequence," in *2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA)*, 2019, pp. 406–409, doi: 10.1109/ICMLA.2019.00076.

[24] I. B. A. Ouahab, M. Bouhorma, L. ElAachak, and A. A. Boudhir, "Proposed precautions for newborn malware family inspired from the COVID19 epidemic outbreak," in *Emerging Trends in ICT for Sustainable Development*, 2021, pp. 53–61, doi: 10.1007/978-3-030-53440-0_7.

[25] M. Venkatasubramanian, A. H. Lashkari, and S. Hakak, "IoT malware analysis using federated learning: a comprehensive survey," *IEEE Access*, vol. 11, pp. 5004–5018, 2023, doi: 10.1109/ACCESS.2023.3235389.

[26] A. Halbouni, T. S. Gunawan, M. H. Habaebi, M. Halbouni, M. Kartiwi, and R. Ahmad, "Machine learning and deep learning approaches for cybersecurity: a review," *IEEE Access*, vol. 10, pp. 19572–19585, 2022, doi: 10.1109/ACCESS.2022.3151248.

[27] X. Jin, X. Xing, H. Elahi, G. Wang, and H. Jiang, "A malware detection approach using malware images and autoencoders," in *2020 IEEE 17th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*, 2020, pp. 1–6, doi: 10.1109/MASS50613.2020.00009.

[28] K. Sethi, R. Kumar, L. Sethi, P. Bera, and P. K. Patra, "A novel machine learning based malware detection and classification framework," in *2019 International Conference on Cyber Security and Protection of Digital Services (Cyber Security)*, 2019, pp. 1–4, doi: 10.1109/CyberSecPODS.2019.8885196.

[29] A. A. Darem, F. A. Ghaleb, A. A. Al-Hashmi, J. H. Abawajy, S. M. Alanazi, and A. Y. Al-Rezami, "An adaptive behavioral-based incremental batch learning malware variants detection model using concept drift detection and sequential deep learning," *IEEE Access*, vol. 9, pp. 97180–97196, 2021, doi: 10.1109/ACCESS.2021.3093366.

[30] D. Wu, P. Guo, and P. Wang, "Malware detection based on cascading XGboost and cost sensitive," *2020 International Conference on Computer Communication and Network Security, CCNS 2020*, pp. 201–205, 2020, doi: 10.1109/CCNS50731.2020.00051.

[31] J. McGiff, W. G. Hatcher, J. Nguyen, W. Yu, E. Blasch, and C. Lu, "Towards multimodal learning for Android malware detection," in *2019 International Conference on Computing, Networking and Communications (ICNC)*, 2019, pp. 432–436, doi: 10.1109/ICCNC.2019.8685502.

[32] N. A. Anuar, M. Z. Mas'ud, N. Bahaman, and N. A. M. Ariff, "Analysis of machine learning classifier in android malware detection through opcode," in *2020 IEEE Conference on Application, Information and Network Security (AINS)*, 2020, pp. 7–11, doi: 10.1109/AINS50155.2020.9315060.

[33] E. Rodriguez, B. Otero, N. Gutierrez, and R. Canal, "A survey of deep learning techniques for cybersecurity in mobile networks," *IEEE Communications Surveys and Tutorials*, vol. 23, no. 3, pp. 1920–1955, 2021, doi: 10.1109/COMST.2021.3086296.

[34] J. Palša *et al.*, "MLMD—a malware-detecting antivirus tool based on the XGBoost machine learning algorithm," *Applied Sciences*, vol. 12, no. 13, 2022, doi: 10.3390/app12136672.

[35] C. T. Thanh, “A study of machine learning techniques for cybersecurity,” in *2021 15th International Conference on Advanced Computing and Applications, ACOMP 2021*, 2021, pp. 54–61, doi: 10.1109/ACOMP53746.2021.00014.

[36] H. Alamro, W. Mtouaa, S. Aljameel, A. S. Salama, M. A. Hamza, and A. Y. Othman, “Automated android malware detection using optimal ensemble learning approach for cybersecurity,” *IEEE Access*, vol. 11, pp. 72509–72517, 2023, doi: 10.1109/ACCESS.2023.3294263.

[37] Y. Wei and Y. Sekiya, “Sufficiency of ensemble machine learning methods for phishing websites detection,” *IEEE Access*, vol. 10, pp. 124103–124113, 2022, doi: 10.1109/ACCESS.2022.3224781.

[38] S. Sharma, N. Gupta, and B. Bundela, “A GWO-XGBoost machine learning classifier for detecting malware executables,” in *2023 International Conference on Disruptive Technologies (ICDT)*, 2023, pp. 247–251, doi: 10.1109/ICDT57929.2023.10150993.

[39] N. Mohapatra, K. Shreya, and A. Chinmay, “Optimization of the random forest algorithm,” in *Advances in Data Science and Management*, vol. 37, 2020, pp. 201–208, doi: 10.1007/978-981-15-0978-0_19.

[40] M. E. Narayanan, “Malware classification using XGBoost with vote based backward feature elimination technique,” *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, vol. 12, no. 10, pp. 5915–5923, 2021, doi: 10.17762/turcomat.v12i10.5412.

[41] S. Guan and W. Li, “EnsembleDroid: A malware detection approach for Android system based on ensemble Learning,” in *2022 IEEE MIT Undergraduate Research Technology Conference (URTC)*, 2022, pp. 1–5, doi: 10.1109/URTC56832.2022.10002213.

[42] N. Pachhala, S. Jothilakshmi, and B. P. Battula, “Prediction of novel malware using hybrid convolution neural network and long short-term memory approach,” *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 14, no. 4, pp. 4508-4517, 2024, doi: 10.11591/ijece.v14i4.pp4508-4517.

[43] R. K. Dubey, N. Dandotiya, A. Sharma, S. Mishra, and S. K. Gupta, “Cyber attack detection using machine learning techniques,” in *2023 IEEE International Conference on ICT in Business Industry & Government (ICTBIG)*, 2023, pp. 1–6, doi: 10.1109/ICTBIG59752.2023.10456080.

[44] Z. Sawadogo, J.-M. Dembele, G. Mendy, and S. Ouya, “Android malware detection: An in-depth investigation of the impact of the use of imbalance datasets on the efficiency of machine learning models,” in *2023 25th International Conference on Advanced Communication Technology (ICACT)*, 2023, pp. 1460–1467, doi: 10.23919/ICACT56868.2023.10079245.

BIOGRAPHIES OF AUTHORS

Grace Odette Boussi obtained her Bachelor of Computer Application in Haryana, India, in 2016. She then pursued a Master of Science in networking technology and management at Amity University Noida from 2016 to 2018, where she received the silver medal for her academic achievements. Since 2019, she has been pursuing her Ph.D. in cyber security at Amity University Noida. She can be contacted at email: graceboussi@gmail.com.

Dr. Himashu Gupta is a respected senior faculty member at Amity University in Uttar Pradesh, India. He completed his education at Aligarh Muslim University and has an extensive academic and professional background in information technology. He has published numerous research papers and articles in the field, with his first patent in network security being published in the international journal of patents by the Government of India in December 2010. Additionally, he is a member of various prestigious international technical and research organizations and has delivered online lectures to students from 16 African countries. He can be contacted at email: hgupta@amity.edu.

Syed Akhter Hossain is an esteemed computer scientist, educator, columnist, and technology consultant from Bangladesh. He is currently serving as a professor and the head of the Department of Computer Science and Engineering at the University of Liberal Arts Bangladesh. He can be contacted at email: aktarhossain@daffodilvarsity.edu.bd.