FUDMA Journal of Sciences (FJS)
ISSN online: 2616-1370
ISSN print: 2645 - 2944
Vol. 10 No. 22, November, 2026, pp 125 – 132
DOI: https://doi.org/10.33003/fjs-2026-1022-6156

Development of a Machine Learning-Based Model for Phishing Attack Detection in Smart-Home Environments

***Ayobami T. Olusesi, Oluwabukunmi B. Onasanya, Oyinkansola E. Deji-Agboola, Ayo Isaac Oyedeji and Olatilewa R. Abolade**

Department of Computer Engineering, Olabisi Onabanjo University, Ogun State, Nigeria.

*Corresponding authors' email: olusesi.ayobami@oouagoiwoye.edu.ng

Received: 04 Oct. 2026
Accepted: 19 Oct. 2026
Published: 24 Oct. 2026

ABSTRACT
Internet of Things (IoT)-powered smart home solutions are increasingly prevalent. As a result, they are significantly altering how we live today. Due to device heterogeneity, resource constraints, user awareness, and other factors, the systems that connect the Internet of Things (IoT) are themselves connected to other devices, making phishing attacks one of the most significant hazards in smart homes. Phishing undermines the privacy, security, and integrity of smart ecosystems by exploiting a user's trust through phony emails, voice squatting, URLs, among others. Since the current security techniques are insufficient, this study aim is to create a new tool based on machine learning (ML) and deep learning (DL) techniques to check and detect phishing attacks on smart home systems. This study makes use of enron dataset from Kaggle which contains a total of 33,715 email samples. Data pre-processing was carried out to extract relevant features and attributes from the dataset. The phishector.py script provides a command line interface for processing emails and obtaining predictions. The model was evaluated using accuracy, precision, recall, and FI-score as performance metrics. From the obtained results, bagged decision tree model shows a 99.94% accuracy, precision 99.8%, F1-score, 99.94%, and recall, 100% while Adobos a 97.98% accuracy, precision 96.23%, F1-score, 98.08%, and recall, 100%. The study's result has further demonstrated how the suggested methods can be applied to detect phishing assaults in smart homes and shield linked smart devices from cyberattacks.

Keywords: Smart Homes, Phishing, Machine Learning, Deep Learning

INTRODUCTION

Over the past few decades, there has been a substantial advancement in the ability of networked devices across numerous users and industries to access and use the internet. These days, practically every modern activity involves the use of Internet-related services. In recent years, human lifestyles have become increasingly dependent on Internet of Things (IoT) services. This technology is being used to create a fully automated environment, or at the very least, efforts are being made to do so (George et al., 2024; Omar et al., 2024). IoT is a solution that makes it possible for people and things to be continuously and globally connected using any network or service (Zorgati et al., 2022). It was initially used to automate individual homes for more comfort and convenience, and it is now being expanded to build smart cities. The use of Internet of Things (IoT) devices to remotely control and automate household appliances and other equipment in order to enhance residents' quality of life is known as smart home technology. It is made up of smart devices connected by a router, including a smart thermostat, lights, security camera, Google Home, Alexa from Amazon, and smart locks. (George et al., 2024). In addition to encouraging environmental cleanliness and helping to create intelligent, sustainable communities, smart homes have a big impact on environmental sustainability. Increased comfort, a variety of energy sources, and housing options that are available to everybody are advantages of these technologies (Tetteh, & Amponsah, 2020). The latest wave of home automation is about more than just convenience. Smart homes can increase energy efficiency and reduce utility costs by automating tasks like lighting, temperature management, and security systems. However, due to continuous technology improvements, smart houses are anticipated to have a significant impact on the future of comfortable and efficient living (Allifah & Zualkernan, 2022).

While smart homes offer convenience and automation, there are significant security and integration challenges. Smart homes can be complex systems with many devices, sensors, and services. This complexity may make it difficult to monitor security and identify intruders (SH2 study). Cyberattacks continue to be largely successful due to human mistake and home users' lack of awareness of security threats in smart homes (Gaber et al., 2024). Effective smart home security requires collaboration between internet service providers, device manufacturers, and homeowners. A 2022 survey found that 40.3% of smart homes worldwide had five or more internet-connected devices, and 40.8% of families had at least one item that may harm the entire house. The majority of house users may not be aware of the cyber threats connected to smart-home devices or may not know how to properly protect them. Users are now viewed as one of the weak points in the cyber risk chain as a result of this (Allifah & Zualkernan, 2022). Because many consumers lack the knowledge required to correctly configure IoT devices, weak passwords, unpatched firmware, out-of-date software, and other security weaknesses are common entry points for attackers (Wan et al., 2020).

A significant risk is posed by the security issues present in smart home environments, such as the usage of weak default passwords, unencrypted data transmission, sporadic firmware updates, and vulnerabilities in third-party applications. Regulatory loopholes further make matters worse because

there are no uniform security standards for IoT devices like smart homes, which leads to varying levels of protection among manufacturers and exposes numerous systems. One of the primary problems with smart home devices is the absence of clear security guidelines from manufacturers. It might be difficult to ensure that smart home gadgets can all operate securely together because they are often made by multiple manufacturers and use different communication protocols (George et al., 2024). The authentication and encryption methods used to thwart infiltration attempts are unable to keep up with the fast changes in security protocols made by various vendors. Other systems are susceptible to security breaches if a smart home system has been exploited (Trim & Lee, 2023) The fact that there are currently no established security standards for smart home devices and that security features might differ greatly between manufacturers and products is another key security concern. Smart home systems are now a threat to data security, privacy, and an increase in attackers due to all of the aforementioned issues. As new security flaws emerge, smart homes have grown more unstable and susceptible (Kavallieratos et al. 2019).

In order to protect smart home systems from hackers and enhance system dependability, a powerful and reliable intrusion detection system is therefore required. Several reviews and intrusion detection techniques have been carried out in the past years. (George et al., 2024). provided extensive survey that gives a full picture of security issues in IoT-enabled smart homes. Its main contribution is a dual-focused research that details the particular vulnerabilities of common smart home devices and looks at threats across the typical four-layer IoT architecture. The process, which synthesizes a variety of risks and countermeasures, is a conventional literature review. Important conclusions include the necessity of layered defenses, the dangers of device heterogeneity, and the crucial role of user awareness. By incorporating cybersecurity awareness into the Trust in Technology Model (TTM), (Alshammari & Al-Mamary, 2025) investigates the factors influencing the adoption of AI-driven smart home technologies. Using a quantitative survey and structural equation modeling, the study concludes that a user's overall inclination to trust technology enhances institution-based trust, which in turn fosters trust in particular devices. The user's cybersecurity awareness and this particular trust both greatly increase the intention to adopt these devices, and this intention is a strong predictor of deep, advanced usage. In order to accelerate the adoption of smart home technologies, the authors suggest that building confidence through institutional assurances and raising public cybersecurity knowledge are essential tactics. They provide important insights for regional politicians and technology innovators.

Schiller et al., (2023*)* developed an approach to improving IoT network security using autonomous, swarm-based penetration testing, in which IoT devices proactively check other devices in the same network for weaknesses. The authors compared a conventional linear penetration testing algorithm with two swarm-based methods—a queue-based algorithm and a nature-inspired Particle Swarm Optimization (PSO) algorithm—across both small smart home (30 devices) and large smart building (250 devices) scales using a custom network simulator (CyberSim-SwarmIoT). The findings showed that swarm-based techniques, especially the PSO algorithm, found vulnerabilities much more quickly and thoroughly than linear techniques. Over time, the PSO approach achieved the highest unique vulnerability detection rate while outperforming the queue-based method in terms of computational efficiency. The study concludes that using coordinated swarms of IoT devices for autonomous penetration testing can significantly increase security awareness and vulnerability identification in smart environments, providing a proactive defense mechanism that makes use of already-existing network resources. With an emphasis on smart wall pads—central control devices with displays that retain substantial user data, Kim et al., (2024) developed a four-stage forensic methodology for smart home environments. Data gathering by network packet capture, vulnerability exploitation, serial port access, and chip-off techniques, data analysis to extract user-related artifacts, data validation for legal proof, and forensic preparation to identify device specs are all part of the methodology. By successfully extracting forensic artifacts like user phone numbers, linked device lists, visitor video thumbnails, security event logs, and voice memos, the authors validate their method on testbeds from three vendors (Samsung, Kocom, and Commax). This shows that wall pads are rich sources of digital evidence and that giving their analysis priority can improve efficiency in smart home forensic investigations.

The important security issues of node capture attacks in IoT-based smart home systems was proposed by (Asghar et al., 2023). The authors first do a security analysis of current user-authentication schemes, finding that many of them are susceptible to such physical attacks and lack crucial security features. They respond by putting up a brand-new, lightweight user-authentication system made specially to fend off node capture assaults. The suggested method includes a number of security characteristics, including forward secrecy, mutual authentication between the user, gateway, and smart devices, user anonymity through the use of dynamic temporary identities, and protection against location monitoring. Formal techniques, such as BAN logic to demonstrate mutual authentication and session key agreement and the ProVerif tool to confirm the confidentiality of key parameters and authentication events, are used to thoroughly validate the security of the suggested scheme. Additionally, compared to a number of cutting-edge protocols, a performance analysis shows that the scheme is computationally and communication ally economical, making it appropriate for IoT devices with limited resources. The study concludes that the suggested plan effectively reduces the vulnerabilities found while keeping overhead to a minimum.

Javed et al., (2024) *proposed* a revolutionary, decentralized intrusion detection system (IDS) that is integrated straight into a smart thermostat. The implementation and quantization of a Cat Boost machine learning model for on-device intrusion detection, a technique not natively supported by existing microcontroller machine learning libraries; the creation of a unique dataset (IDSH) based on features extractable directly from a microcontroller (ESP32) using the lip library, as opposed to traditional gateway-captured traffic; and a thorough performance evaluation contrasting the suggested Cat Boost-based IDS against an XGBoost-based IDS are some of the major contributions. In terms of accuracy, inference speed, and program storage efficiency, the system beat the XGBoost benchmark, demonstrating the viability of directly implementing a reliable, lightweight IDS on resource-constrained IoT devices to improve protection against frequent intrusions. Kavallieratos et al., (2019) offered a thorough methodology for threat analysis and modeling. The main component of their strategy is utilizing an existing smart home reference architecture to conduct a thorough threat analysis using Microsoft's threat modeling tool and the STRIDE method, concentrating on the communication and physical perspectives across six smart home scenarios of varying complexity. The investigation

showed that IoT devices, particularly those with transitive connections like IP cameras, greatly expand the attack surface and that more complicated topologies inherit vulnerabilities from simpler ones. Okoro et al. (2026) developed devised a TinyML-optimized multimodal LSTM for unsupervised anomaly detection in smart homes. The study amalgamates multimodal sensing, attention-driven temporal modeling, unsupervised reconstruction, and an adaptive mechanism into a unified experimental framework.

However, as the IoT ecosystem continues to grow, intrusion such as phishing attack needs more attention in order to protect user data, privacy, and prevent more significant issues (Deldar & Abadi 2023). Phishing is a gift that may cause significant harm, and the proliferation of smart home technologies has significantly increased their vulnerability to cyber threats. Phishing using voice command or email takes advantage of user trust (Gupta et al., 2018). In order to obtain unauthorized access to smart devices or their networks, attackers pose as reliable parties. Attackers might, for instance, utilize voice squatting to steal data from unprotected devices or activate home assistants with similar-sounding utterances (Zhang et al., 2021) Therefore, for threat avoidance and suitable intrusion detection techniques, new techniques and outcomes are essential. Recently developed machine learning (ML) and deep learning (DL) techniques are being used for intrusion detection, categorizing anomalies in the network, and discouraging it. Convolutional neural networks (CNNs), deep belief networks (DBNs), recurrent neural networks (RNNs), and restricted Boltzmann machines (RBMs) are among the networks found in DL (Sabeenian et al., 2020; Almutairi, & Alshargabi, 2022). But, the security measures in place are insufficient. The authentication and encryption methods used to stop phishing assaults are unable to keep up with the advancements. Additionally, IoT devices lack the resources necessary to combat phishing assaults. In multi-device linked environments, traditional intrusion detection systems (IDSs) are unable to keep up with cybercriminals' rapid-fire attacks (Trim & Lee, 2023). The lack of uniform security procedures across manufacturers is one of the main issues with smart home appliances. An advanced solution that can use ML and DL to detect phishing assaults on smart home systems in real time and turn them into anti-phishing systems is needed. Hence, this study aims to develop and evaluate a machine-learning-based phishing detection model for smart-home environments.

MATERIALS AND METHODS

Using machine learning (ML) and deep learning (DL) techniques, this study developed a dedicated tool to identify phishing attempts in smart home systems. In order to effectively identify phishing attempts across heterogeneous IoT devices, the methodology combines supervised and unsupervised learning algorithms to assess data from network traffic, device logs, and user interactions. While unsupervised DL models, like auto encoders, use anomaly detection to identify new threats, supervised ML models, like random forests, will categorize existing phishing trends. Lightweight model architectures and edge computing were used to support the resource limitations of IoT devices, guaranteeing interoperability with various protocols like Zigbee and Wi-Fi. Federated learning and other privacy-preserving strategies will be used to reduce data exposure and boost user confidence. Data from simulated smart home environments was gathered, models are trained and optimized, and performance is assessed using metrics like accuracy and false positive rate. Virtual machine (VM) running Kali Linux, an open-source Linux distribution for penetration testing primarily used by cybersecurity experts and black hats, to simulate a phishing attack scenario in order to gain a better understanding of how phishing attacks operate.

Phishing Detection Tool

Using Python libraries such as numpy for numerical computation, pandas for data manipulation, BeatifulSoup for HTML parsing, scikit-learn for machine learning models, and TensorFlow for the deep learning model, and four the phishing detection tool were used which are listed below:

i. *Kali Linux VM:* A virtual machine hosting a 2023 version of the Kali Linux operating system.
ii. *Koadic:* An open-source windows operating system post-exploitation tool installed using the Kali Linux VM terminal with the command *sudo apt install Koadic*.
iii. *Two Dummy Gmail Accounts:* Two Gmail accounts were created, one to function as the victim's email address and the other to function as the attacker's email address.
iv. *Network:* The VM was configured using an NAT bridged adapter network to simulate connectivity with the VM and base windows OS as two different machines with different IP addresses.

The Phishing Simulation was Carried out in the Following Steps

i. After creating two fictitious Gmail accounts to act as the victim's and attacker's email addresses, a Kali Linux operating system was installed on Oracle's virtual box, a well-known free virtual machine.
ii. In order to trick the victim into downloading my payload and granting access to their computer, an Apache server running the command sudo apt install apache2 was used to host a dummy webpage. HTML and CSS were utilized to create a fake smart-home company webpage.
iii. A payload named 'security_update.bat' to echo the IP address and the port where stager is hosted on the Kali Linux VM and both my dummy webpage and the payload were hosted on the apache server.
iv. A straightforward phishing email sample was created using the attacker's email account to route the victim to my hosted fictitious webpage. Google Chrome's safe browsing function and Windows Security Defender were turned off to enable the download of dubious files.

The Model Architecture

The basic idea behind hybridization is to improve detection performance by using machine learning to choose meaningful characteristics and/or classify the produced data and deep learning to automatically extract complicated features from phishing-related data. The following processing steps provide an explanation of the model's design as depicted in Figure 1.

Data Collection

Gathering information from the smart home network and the digital services that its residents utilize is the first step. The home router or IoT gateway can be used to monitor pertinent network traffic. The data used for this study was the enron dataset from Kaggle which contains a total of 33,715 email samples in a .csv format with four columns namely "Message", "Subject", "Spam/Ham" and "Date". The "Message" column contains the full content of the email sample, including its headers, body, and metadata, while the

"Spam/Ham" column contains values 'ham' for ham emails and 'spam' for spam emails.

Data Preprocessing

Noise, missing values, duplicate entries, and mismatched feature formats could all be present in the gathered data. The raw data is transformed into an appropriate representation for the learning algorithms through preprocessing. Data preprocessing was carried out to extract key relevant features and attributes from the dataset. The phishector.py script implements these functions.

Deep Learning Base Feature Extraction

This is the hybridization process's first significant step. Deep learning uses raw or minimally processed data to automatically identify complicated patterns. Deep learning can find associations that could be challenging to articulate using straightforward rules, in contrast to traditional feature engineering, which mostly relies on manually defined features.

Machine Learning Base Feature Extraction

The system employs machine learning to find the most informative features once deep learning has extracted meaningful representations

Model Development

The phishing detection model was created using with the following approach:

i. *Input Layer:* This layer accepts 40 features, matching the features extracted by the detection tool script.
ii. *Output Layer:* The model provides a probability of the email being classified as either spam or ham.
iii. *Optimizer:* The model implements Adam with a learning rate of 0.001.

The model was trained on the enron spam dataset with a training process involving 2o epochs with a batch size of 32 in the *create_model.py* python script. These models were trained and saved as pickle files to be used by the tool for email classification. The Phishector.py script provides a command line interface for processing emails and obtaining predictions. Key implementation details include the input for user directory containing email files as shown in Figure 2 and the menu interface which is made up of four steps as shown in Figure 3

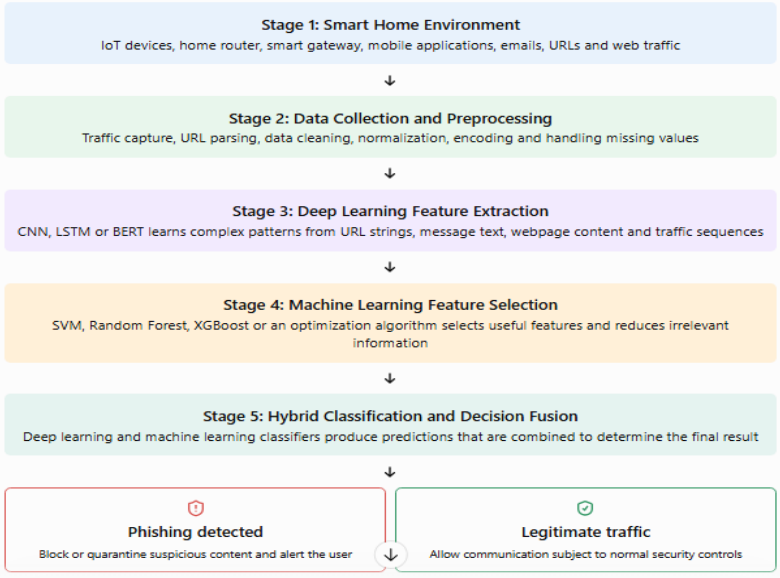

Figure 1: The Hybridized ML/DL Model Architecture of Phishing Detection in Smart Homes

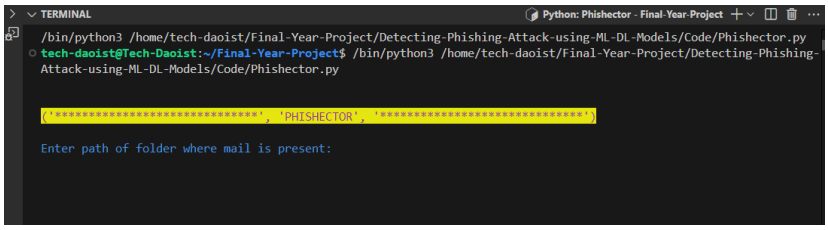

Figure 2: Input Interface of Tool

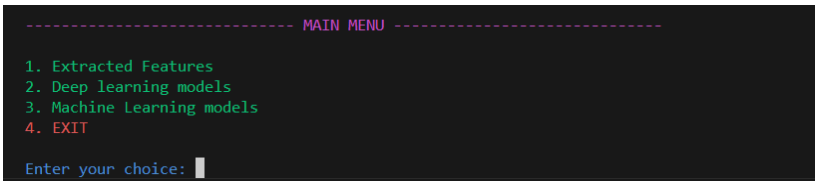

Figure 3: Menu Interface

RESULTS AND DISCUSSION

Bagged Decision Tree, Random Forest, Extra Trees, AdaBoost, Stochastic Gradient Boosting (SG Boosting), Naive Bayes, Support Vector Machine (SVM), and Voting Ensemble were the eight machine learning models that were assessed. Accuracy, precision, recall, F1-score, and area under the Receiver Operating Characteristic curve (AUC-ROC) are the evaluation metrics that were employed. Table 1 depicts the table summarizing the various evaluation metrics of the eight machine learning models and the deep learning model. The heatmap in Figure 4 provides a summary of the metrics with the darker shades of color indicating higher values.

Table 1: Evaluation Metrics of the Various ML and DL Models

Model	Accuracy	Precision	Recall	F1-Score	AUC-ROC
Bagged DT	0.999406	0.998848	1	0.999424	0.999308214
Random Forest	0.9994069	0.998848	1	0.999424	0.999308214
Extra Trees	0.9994069	0.998848	1	0.999424	0.999375328
AdaBoost	0.9798339	0.962264	1	0.980769	0.992325393
SGB	0.9994069	0.998848	1	0.999424	0.999758476
Naive Bayes	0.5247628	0.521625	0.914648	0.664363	0.71494018
SVM	0.815688	0.772071	0.910323	0.835517	0.880161661
Voting Ensemble	0.9994069	0.998848	1	0.999424	0.999771018
Deep Learning	0.9991103	0.998273	1	0.999136	0.999322473

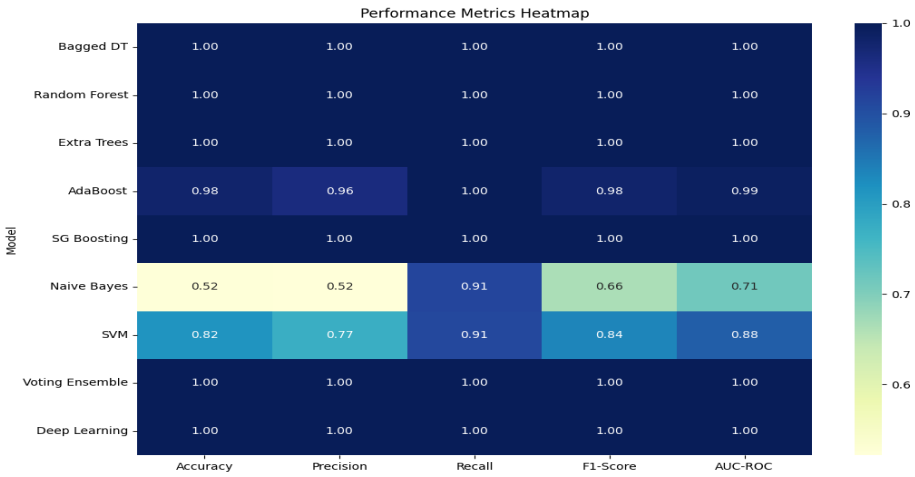

Figure 4: Evaluation Metrics Heat map

All models except the Naive Bayes and the Support Vector Machine model got a similar accuracy percentage (100% ~ 98%). Both models also had a lower F1-Score, precision and accuracy compared to the rest of the models. The ROC curves as shown in Figure 5 demonstrate the trade-off between the recall (true positives) and false positive rates for all the Machine Learning models and the deep learning model. The balance between precision and recall is highlighted by the position recall curves shown in Figure 6. It demonstrates that every machine learning model aside from Naive Bayes and Support Vector Machine maintains a high degree of precision even when recall rises, and the deep learning model does the same.

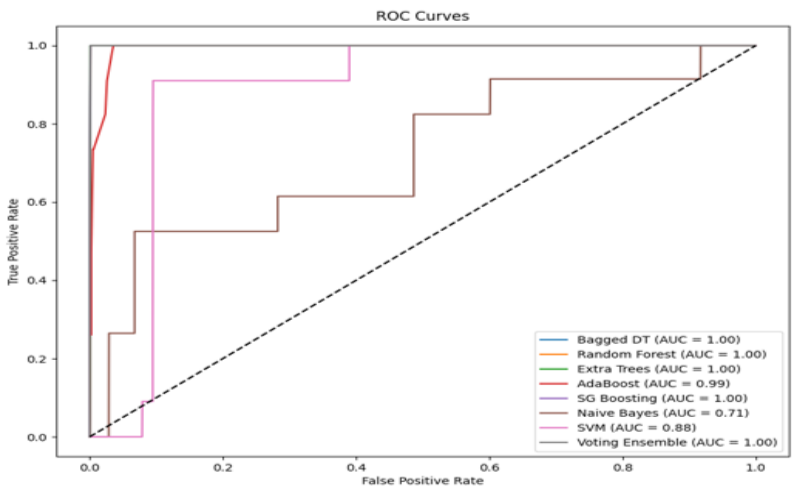

Figure 5: ROC Curve Across All ML Models

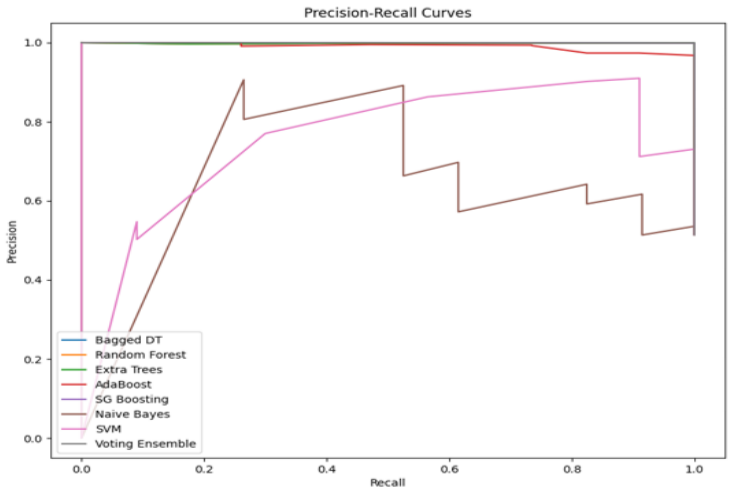

Figure 6: Precision-Recall Curve

Figure 7 shows the confusion matrix for AdaBoost classification method which is one of the machine leaning techniques and it was used to differentiate between two classes: Phishing (malicious traffic/messages) and Ham (legitimate traffic/messages). The matrix contrasts the predicted label (horizontal axis) with the true/actual label (vertical axis). Assuming Phishing is the positive class:

i. True Negative (TN) = 3,140: Legitimate samples correctly classified as Ham.
ii. False Positive (FP) = 136: Legitimate samples incorrectly classified as Phishing
iii. False Negative (FN) = 0: Phishing samples incorrectly classified as Ham.
iv. True Positive (TP) = 3,468: Phishing samples correctly detected.

There are 6,744 test samples in all, of which 3,276 are Ham samples and 3,468 are Phishing samples. Out of 6,744 samples, 6,608 were correctly identified by the model. This suggests that the test set's overall classification performance was strong.

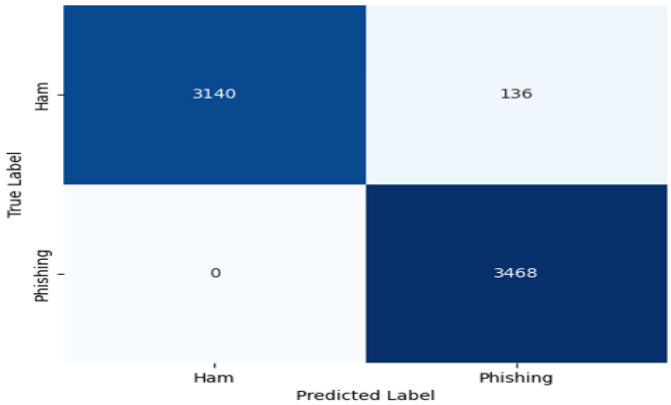

Figure 7: Confusion Matrix of Adaboost Model

All of the phishing samples in this test set were identified by the model. Missed phishing attacks can put users and smart-home systems at danger, making this very useful in cybersecurity. 96.23% of the samples that were categorized as phishing were actually phishing. The remaining 3.77% were authentic samples that were mistakenly reported as harmful. The most notable observation is that AdaBoost detected all 3,468 phishing samples in this test set, with no false negatives. However, it incorrectly flagged 136 legitimate samples as phishing. AdaBoost obtained a 98.08% F1-score, 100% recall, 97.98% accuracy, and 96.23% precision. These findings show that the assessed dataset has a potential phishing detection capability, with false-positive reduction being a crucial topic for future development. The confusion matrix in Figure 8, shows the true label on the vertical axis and the predicted label on the horizontal, assesses how well the Bagged Decision Tree (Bagged DT) classifier performs in differentiating between two classes: Phishing (malicious samples) and Ham (legitimate materials).

i. True Negative (TN) = 3,272: Legitimate samples correctly classified as Ham.
ii. False Positive (FP) = 4: Legitimate samples incorrectly classified as Phishing.
iii. False Negative (FN) = 0: Phishing samples incorrectly classified as Ham.
iv. True Positive (TP) = 3,468: Phishing samples correctly identified.

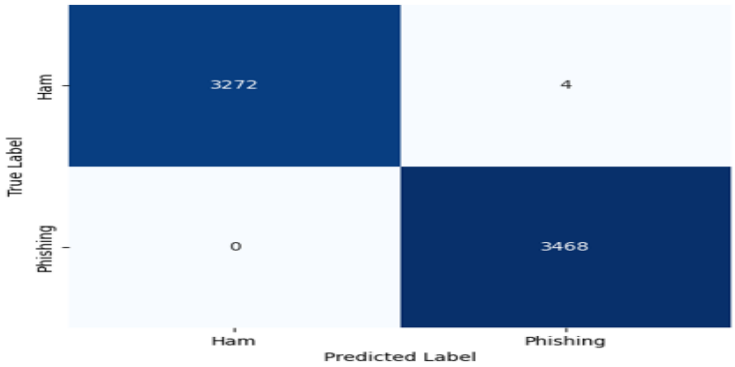

Figure 8: Confusion Matrix of Bagged Decision Trees Model

On this test set as shown in Figure 8, the model performed exceptionally well overall, correctly classifying 6,740 out of 6,744 items. All of the phishing samples in this test set were identified by the model. Missed phishing attacks can put users and smart-home systems at danger, making this very useful in cybersecurity. 96.23% of the samples that were categorized as phishing were actually phishing. The remaining 3.77% were real samples that had been mistakenly reported as harmful. These findings indicate that Bagged DT is very good at differentiating phishing from authentic samples for a study on phishing attack detection in smart homes. Bagged DT shows a 99.94% F1-score, 100% recall, 99.94% accuracy, 99.88% precision, false positive rate of 0.12% and false negative rate of 0.00%. For smart-home security, its performance is very encouraging because every phishing sample in the assessed dataset was identified. Only four false alerts were produced, which decreased the possibility of needless blockage of valid connections. Reliable categorization on the test set is shown by its high accuracy and specificity. When compared to a single decision tree, its ensemble structure can increase robustness by combining predictions from several decision trees.

Conclusion

It is well known that phishing attack causes significant harm in digital and web-based devices. The proliferation of smart home technologies has significantly increased their vulnerability to cyber threats. Phishing using voice command or email takes advantage of user trust in order to obtain unauthorized access to smart devices or their networks, attackers pose as reliable parties especially to users who frequently use a smartphone or voice command to engage with their smart home functionality. However, because smart gadgets with little processing power are unable to implement strong security protocols, this connectivity leads to vulnerabilities. This study has clearly shown that the use of both deep learning and machine learning as a hybrid approach for phishing attack detection is needed in order increase the security domain of smart home devices and reduce the vulnerability of the users to cyber-attacks. From the machine learning techniques, it clearly shows that Bagged DT performed better than Adaboost Model, however making decisions on deployment, data leak detection, and assessment of new phishing attempts are still open for further investigations. Additionally, using a metaheuristic optimization strategy during the feature selection stage might further enhance the performance of the machine and deep learning models.

Acknowledgment

The authors take full responsibility for the publication's content. There are no conflicts of interest disclosed by the writers.

REFERENCES

Allifah, N.M., & Zualkernan, I.A. (2022).

Ranking security of IoT-based smart home consumer devices. *IEEE Access*, 10, 18352–18369.

Almutairi, A.F., & Alshargabi, A.A. (2022).

Using Deep Learning Technique to Protect Internet Network from Intrusion in IoT Environment. *2nd International Conference on Emerging Smart Technologies and Applications (*eSmarTA) (pp. 1 6). IEEE.

Alshammari, M.M., & and Hasan Al-Mamary,

Y. (2025). Building Trust and Cybersecurity Awareness in Saudi Arabia: Key Drivers of AI-Powered Smart Home Device Adoption. *Systems*, 13, 863, 1 – 30. https://doi.org/10.3390/systems13100863

Asghar, I., Muhammad, A.K., Tahir, A.,

Subhan, U., Khwaja, M., & Attaullah, B. (2023). Fortifying Smart Home Security: A Robust and Efficient User-Authentication Scheme to Counter Node Capture Attacks. *Sensor*, 23, 7268, 1 – 18. https://doi.org/10.3390/s23167268

Deldar, F., & Abadi, M. (2023). Deep learning

for zero-day malware detection and classification: A survey. *ACM Comput. Surv.* 56, 1–37.

Gaber, M.G., Ahmed, M., & Janicke, H. (2024). Malware detection with artificial intelligence: A systematic literature review. *ACM Comput. Surv*. 56, 552.

Gupta, B. B., Arachchilage, N. A., & Psannis,

K. E. (2018). Defending against phishing attacks: Taxonomy of methods, current issues and future directions. Telecommunication Systems. 1 – 32.https://doi.org/10.48550/arXiv.1705.09819

George, V., George, H., Eleftheria, K., & Nikos, P. (2024). Review of Smart-Home Security Using the Internet of Things. *Electronics*, 13, 1 – 35. https://doi.org/10.3390/electronics13163343

Javed, A., Muhammad, N.A., Ayyaz-ul-Haq, Q., Muhammad, J., Jehangir, A., & Hadi, L. (2024). Embedding Tree-Based Intrusion Detection System in Smart Thermostats for Enhanced IoT Security. Sensors 24, 7320, 1 – 15. https://doi.org/10.3390/s24227320

Kavallieratos, G., Chowdhury, N., Katsikas, S., Gkioulos, V., & Wolthusen, S. (2019). Threat analysis for smart homes. *Future Internet*, 11, 207

Kim, S., Jewan, B., & Taeshik, S. (2024). Forensic Analysis for Cybersecurity of Smart Home Environments with SmartWallpads. *Electronics*, 13, 2827, 1 – 21.

Kavallieratos, G., Chowdhury, N., Sokratis, K., Vasileios, G., & Wolthusen, S. (2019). Threat Analysis for Smart Homes. *Future Internet*, 11, 207, 1 – 19. https://doi:10.3390/fi11100207.

Okoro, E.O., Ibrahim, N.M., Kirubakaran, P., & Muhammad, S.A. (2026). TinyML-Optimised Multimodal Lstm for Unsupervised Anomaly Detection in Smart Home IoT Systems. FUDMA Journal of Sciences, 10 (20), 145 – 156. https://doi.org/10.33003/fjs-2026-1020-5988.

Omar. A., Khaled, S., & Usman, B. (2024). Towards Securing Smart Homes: A Systematic Literature Review of malware Detection Techniques and Recommended Prevention Approach. *Information*, 14, 1 – 17. https://doi.org/10.3390/info15100631

Schiller, T., Bruce, C., Annie, S. W.& Sean, M. (2023). Security Awareness in Smart Homes and Internet of Things Networks through Swarm-Based Cybersecurity Penetration Testing. *Information*, 14, 536, 1 – 23. https://doi.org/10.3390/info14100536

Sabeenian, R., Deivanai, N. & Mythili, B. (2020). Wild animal's intrusion detection using deep learning techniques. *Int. J. Pharm. Res*, 12(4),1053-1058.
Tetteh, N., & Amponsah, O. (2020).

Sustainable adoption of smart homes from the Sub-Saharan African perspective. *Sustain. Cities Soc.* 63, 102434.
Trim, P.R., & Lee, Y. (2023). Managing

Cybersecurity Threats and Increasing Organizational Resilience. *Big Data Cogn. Comput.* 7, 1 – 3. https://doi.org/10.3390/bdcc7040177

Wan, Y., Xu, K., Xue, G., & Wang, F. Argos (2020). A multi-layer security monitoring system for Internet-of-Things in smart homes. *In Proceedings of the IEEE Conference on Computer Communications (INFOCOM), Toronto*, 1–10.

Zorgati, H.; Djemaa, R.B., & Amor, I.A.B. (2022). Finding Internet of Things resources: A state-of-the- art study. *Data Knowlegde. Eng.*, 140, 102025

©2026 This is an Open Access article distributed under the terms of the Creative Commons Attribution 4.0 International license viewed via https://creativecommons.org/licenses/by/4.0/ which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is cited appropriately.