Volume 1, 2025, Pages 43-51 *Journal of Computer and Communication Networks*
| Regular Article | Open Access

Machine Learning for Cyber Threat Detection Using Historical Vulnerabilities and Security Standards

Arulmurugan Ramu
Department of Computational Science and Software Engineering, K. Zhubanov University (Heriot-Watt), Aktobe Campus, Aktobe, Kazakhstan.
a.ramu@hw.ac.uk

Article Info
Journal of Computer and Communication Networks
https://www.ansispublications.com/journals/jccn/jccn.html

https://doi.org/10.64026/JCCN/2025005

Received 16 November 2024
Revised from 15 January 2025
Accepted 28 February 2025
Available online 16 March 2025
Published by Ansis Publications

Corresponding author(s):
Arulmurugan Ramu, Department of Computational Science and Software Engineering, K. Zhubanov University (Heriot-Watt), Aktobe Campus, Aktobe, Kazakhstan.
Email: a.ramu@hw.ac.uk

Abstract – Changing modern cybersecurity threats make it necessary for organizations to develop security detection systems that integrate supervised learning capabilities with standard security data and historic logs. We propose adopting a multi-level machine learning framework to classify threats by unifying security data from NVD, CVE, VulDB, CAPEC, ATT&CK and CWE sources. The security system performs pre-processing on data before using Decision Trees, Random Forests, Bagging, Boosting and XGBoost algorithms to build features that strengthen threat detection performance. We chose ensemble learning to achieve optimal predictions in threat assessment. The experimental data demonstrates that XGBoost delivers better results among other tested models since it detects cyber threats with a 97.2% success rate, which shows its capability to detect password attacks, phishing details, SQL injection threats, ransomware and DDoS events. Continuous improvement in the system comes from expert feedback systems, which generate patterns when ranking vulnerabilities. The proposed system presents real-time tracking abilities together with dynamic threat-detection methods that enable organizations to use an advanced system for facing emerging cyber threats.

Keywords – Machine Learning, Cybersecurity, Threat Detection, Vulnerability Analysis, Attack Patterns, Anomaly Detection, Intrusion Prevention, Network Security, Data-Driven Security, AI Security.

I. INTRODUCTION

The term “cyber” denotes networks of data systems infrastructure, commonly identified as “virtual reality.” Cybersecurity safeguards the confidentiality, integrity, and security of communication, integration, life, tangible and intangible assets, and information inside an electronic ecology created by companies, individuals, and institutions in data systems. Briefly, cybersecurity safeguards the integrity of digital existence on cyber systems. The protection of data confidentiality and integrity inside information systems infrastructure falls under the domain of cybersecurity [1]. The principal objective of cybersecurity is to safeguard the data of organizations and individuals online. Unawareness of this critical matter can have significant risks. For example, a someone with nefarious intents can penetrate devices via commandeer data and individuals or expropriate user data, including user ID passwords or credit card details. Such attacks may inflict monetary harm on large organization, small organizations, individuals, and even governmental entities. Recent studies indicate that cyberthreats incur costs amounting to billions of dollars for the worldwide economy.

The frequency and intricacy of threats and attacks are escalating daily, while the means accessible to potential assailants are getting increasingly sophisticated and successful. Consequently, for IoT to realize its full potential, it must be rigorously safeguarded against risks and vulnerabilities. The security threats at each layer vary according to their distinct characteristics. The following outlines security threats and vulnerabilities categorized by tiers. At the perception layer, intelligent sensors and RFID tags autonomously recognize the environment and facilitate data flow among devices [2]. Security concerns represent a significant issue inside the perception layer. In the perception layer, most dangers originate from external entities,

primarily sensors and other data collection equipment. These devices are usually both inexpensive to buy and small in size yet they have weak protection against physical attacks.

To detect contemporary cyber threats correctly one needs to perform proactive threat detection methods. The proactive detection of threats through vulnerability and potential attack vector identification happens before their exploitation takes place. The system uses current data analysis and continued active monitoring and advanced analytics to determine security threats before they occur. Distributed systems need essential proactive threat detection because their diverse data and applications spread across numerous environments and infrastructures. Since dynamic systems operate in real-time security protocols require mechanisms to detect abnormal events and produce threat response recommendations for vulnerability reduction [3]. Businesses today adopt distributed technology at an increasing rate which creates both fresh cybersecurity risks and promising security alternatives. Distributed systems consisting of cloud infrastructures together with microservices along with edge computation and network application systems provide scalability and flexibility with reliable features. These technological advancements result in broadened exposure areas which makes it harder to protect interconnected elements. The current security methods fail to protect systems because attackers develop creative techniques to find security weaknesses. Future information systems need superior security methods beyond basic perimeter protection and reactive monitoring because new developmental features continuously enter the systems.

The emergence and growing prevalence of machine learning have led to several studies proposing ML solutions for various cybersecurity challenges, culminating in hundreds of research articles. This abundance has prompted numerous literature assessments that consolidate or summarize the current state of the art. Nonetheless, the majority of these studies may offer an in-depth examination, albeit focused on a singular application, such as cyber risk assessment or IoT security. Some individuals may concentrate on a particular cyber detection issue, such as malware, spam, or intrusion detection [4]. Certain articles do not expressly concentrate on machine learning, while others do not emphasize cybersecurity. Ultimately, numerous studies focus exclusively on particular machine learning paradigms, including generative adversarial networks, adversarial machine learning, reinforcement learning, or deep learning—the latter of which may not represent the optimal "universal" machine learning solution for cybersecurity.

The main goal of this investigation involves creating an intelligent system for detecting threats through machine learning which evaluates previous vulnerability data with standardized security information. The system design utilizes ensemble learning together with different classification models in order to enhance threat detection precision. The system makes its threat detection functions better through continuous expert feedback integration so it can give real-time protection against modern cyber threats. The remaining parts of the study have been organized in the following manner: Section II reviews related works on ML-based cybersecurity threat detection using historical hazards and data related to cyber security. Section III describes the research model, which highlights (i) data preprocessing and future engineering, (ii) ML model training, and (iii) ensemble learning and model selection. Section IV presents a discussion on the findings of our study and presents a flowchart for threat detection and vulnerability assessment. Lastly, Section V concludes the study and highlights the significance of the ML detection system to assess crime-based vulnerability data.

II. RELATED WORKS

In [5], machine learning techniques have demonstrated their essentiality across all intrusion detection methodologies. Implemented strategies include k-nearest neighbors, support vector machines, and decision trees. Another advantage noted is that models may be updated with new data to sustain their performance, thereby addressing evolving threats. The decision tree-based method proves effective for anomaly detection since it demonstrates capability in recognizing beneficial versus detrimental network activities. SVMs classify secure actions from risky ones through their ability to divide data space using a hyperplane. According to K-nearest neighbors, examining data instances leads to successful intruder detection. Security improvement in networks occurs through multiple machine learning algorithm implementations.

The rule-based models previously used face challenges with adapting to modern cyber threat environments says Ashraf et al. [6]. The technology of reinforcement learning agents grants organizations the capability to develop best security protocols while monitoring new attacks through real-time engagement with network systems. As part of the reinforcement learning-based intrusion detection system the agent needs to make safety decisions in order to optimize future rewards. The agent controls security functions through blocking traffic along with policy updates and direct threats management. The agent builds its operational understanding through experimental methods which it uses to change its work procedures following input from the network system. Reinforcement learning technology demonstrates high operational effectiveness in intrusion detection through its ability to adapt to different environments. Systems based on rules find it challenging to modify their defensive actions as new patterns of cyber dangers present themselves in an ever-changing security domain.

Namjoshi and Narlikar [7] explain that most modern signature-based intrusion detection systems utilize string matching combined with regular expression matching technology for their operations. The metamorphic and polymorphic attacks enable attackers to evade signature-based detection systems that protection systems use for detection. New threats make signature-based Intrusion Detection Systems which depend on string and regular expression matching ineffective in detection functions. Application semantics is employed in the form of vulnerability signatures to identify these complex assaults. A single vulnerability may have multiple potential exploitation. The vulnerability signatures outlined in [8] can identify such attacks and augment the precision and detection efficacy of Intrusion Detection Systems (IDS) by leveraging comprehensive application semantics and protocol awareness. Nayak et al. [9] utilizes the attributes of a vulnerability to create a signature for all exploit versions of that vulnerability prior to any exploit being observed in the wild. The signatures are implemented at the network IDS, enabling it to filter traffic that exploits a vulnerability in a specific host application until the host applies a patch for the vulnerable application. Nevertheless, Lippmann, Webster, and Stetson [10] discovered that no actual

implementation of these methods currently exists in practice. Despite being offered as free software, Net Shield [11] is a prototype solution capable of managing a restricted number of protocols.

According to Lee, Shin, and Realff [12], all machine learning skills must be considered to enhance the models, taking into account elements such as computational time, update potential, and complexity. The priority may differ based on the application. Moreover, many performance metrics are evaluated in addition to the error rate, as alternative indicators have demonstrated numerous advantages. The efficacy of artificial classifiers in accurately detecting malware has been evaluated, addressing false-positive instances with effective outcomes utilizing perceptron-oriented classifiers. Machine learning methodologies have been employed in the creation of threat-detection models, utilizing neural network classifiers, support vector machines (SVM), Bayesian classifiers, Naive Bayes, Bayesian regularized neural networks, self-organizing maps, among others. Azam, Islam, and Huda [13] assert that machine learning methodologies, including SVM and decision trees, can efficiently differentiate between attack internet sessions. Neural networks and fuzzy logic have been effectively integrated for malware discovery, focusing on the most significant API requests.

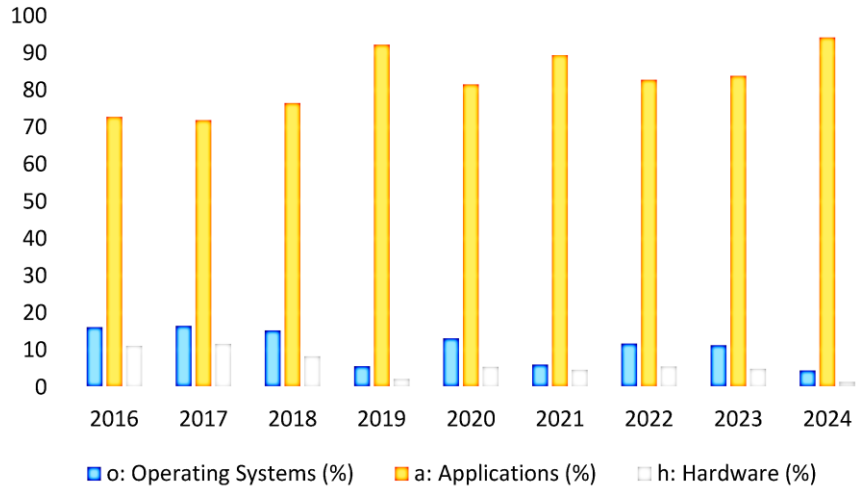

Fig 1. Distributions of CPE Quantities Retrieved from CPE/NVD DICT by Subdivision

As described by Hubballi and Suryanarayanan [14], conventional cybersecurity approaches, which frequently depend on established rules and signature-based detection technologies, are becoming progressively insufficient in this evolving threat environment. Standard security operations implement response steps following threat detection that might lead to property damage because of short response durations. The implementation of methodological restrictions makes it difficult for existing systems to detect threats immediately because large datasets from devices slow down these systems. The current cybersecurity strategies require enhancement to detect modern threats including zero-day attacks alongside APTs because contemporary security needs to identify emerging concerning threats through modern defensive solutions. Big data analytics together with machine learning technology allows modern cybersecurity systems to detect threats quickly before implementing appropriate responses. The combination of security system algorithms with large databases enables unauthorized behavior pattern identification as security threats.

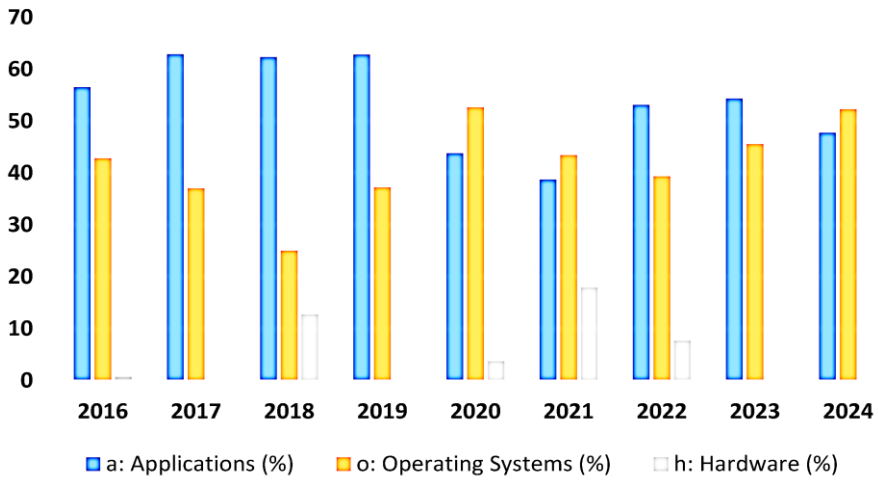

Fig 2. Partitioned Distributions of CPEs Collected Via the CVE/NVD API

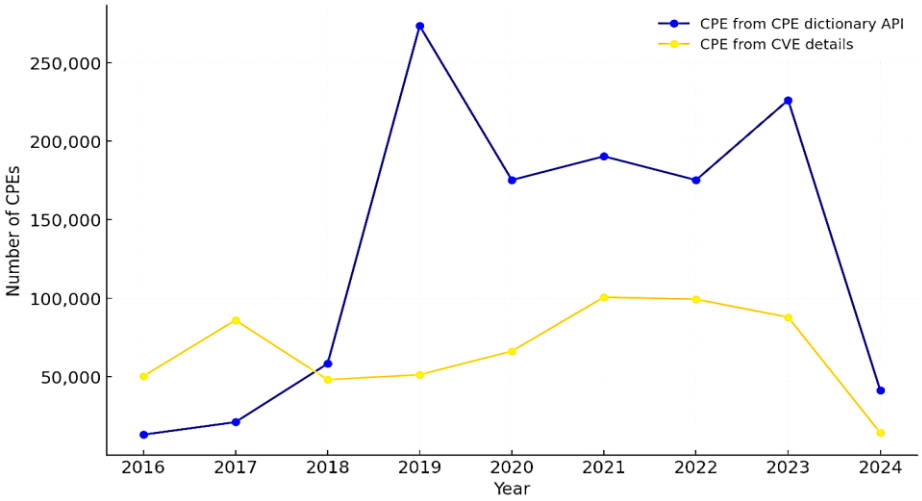

Fig 3. Comparison Between CPEs Removed from NVD

In a report by Benetis et al. [15], CVE (Common Vulnerability and Exposure) program is administered by the MITRE Corporation and is supported by the U.S. DHS (Department of Homeland Security) and the CISA (Cybersecurity and Infrastructure Security Agency). It emphasizes the representation of a terminology and lexicon pertaining to security-related product vulnerabilities. Each CVE ID is allocated to the corresponding product by designated entities referred to as CNAs (CVE Numbering Authorities). The NVD (National Vulnerability Database) oversees the analytical procedure for every CVE ID, including reference tags, the CWE (Common Weakness Enumeration), the CVSS (Common Vulnerability Scoring System), and CPE Applicability Statement. The annual publication of CVEs by the NVD consistently rises. **Fig. 1, 2, 3,** and **4** present statistics for supplementary CPE data from 2016 to 2024. This underscores a significant disparity between CPEs published with metadata/CVE and those enumerated within the dictionary. Moreover, it is essential to acknowledge that not all CPEs impacted by published threats are included in each CVE entry.

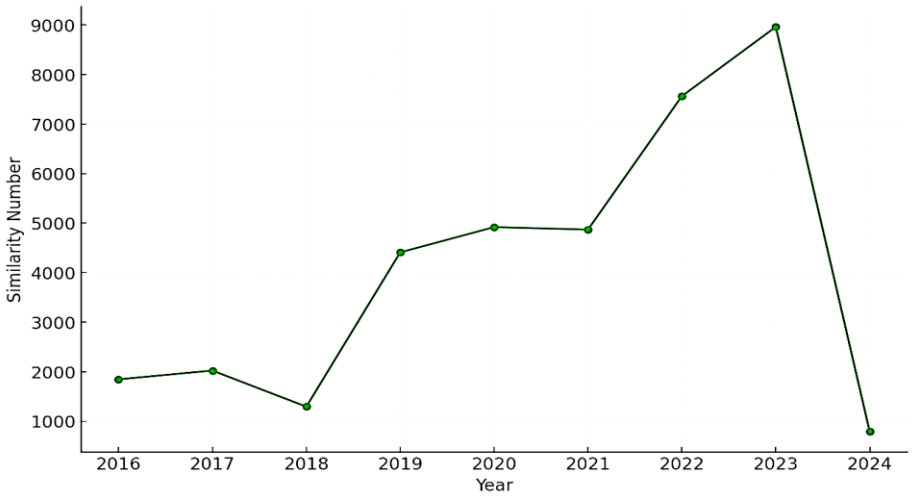

Fig 4. Comparison CPEs Rate between Dictionary/NVD and CVE/NVD.

III. RESEARCH MODEL

A framework managed by machine learning uses vulnerability database records with security standard protocols to detect threats. The model structures different interconnected layers to process threats before analysis results in threat classification.

Data Preprocessing and Feature Engineering

Data cleaning and normalization together with feature extraction occur during the preprocessing stage for input dataset X which comprises security events represented by samples x_i with n features. Let $X = x_1, x_2, \ldots, x_m$ be the input feature space. The normalization process requires Min-Max scaling using Eq. (1).

$$x' = \frac{x - \min(X)}{\max(X) - \min(X)} \quad (1)$$

The normalization feeds the x' value into the system to normalize features which limit values between 0 to 1. Massive label encoding accompanies the mutual information measure $I(X;Y)$ as it selects features while utilizing target labels as Y.

The classification accuracy receives an enhancement from dimensional reduction achieved through retention of the most important features computed using Eq. (2).

$$I(X;Y) = \sum_{x\in X}\sum_{y\in Y} P(x,y)\log\frac{P(x,y)}{P(x)P(y)} \quad (2)$$

Machine Learning Model Training

XGBoost and 4 other classification models consisting of Decision Trees, Random Forests, Bagging, boosting together with XGBoost undergo training within the system (see **Table 1** and **2**). The function $f: X \to Y$ transforms elements in X to predicted $\hat{Y}$ values. Cross-entropy is used in Eq. (3) to compute L when optimizing the model parameters:

$$L = -\sum_{i=1}^{m}[yi\log(\hat{y}i) + (1-yi)\log(1-\hat{y}i)] \quad (3)$$

The actual label y_i meets the predicted probability $\hat{Y}i$. The evaluation combines accuracy and precision with recall and F1-score computation in Eq. (4).

$$\text{Accuracy}=\frac{TP+TN}{TP+TN+FP+FN};\text{Precision}=\frac{TP}{TP+FP} \quad (4)$$

$$\text{Recall}=\frac{TP}{TP+FN};\text{F1score}=2\times\frac{\text{Precision}\times\text{Recall}}{\text{Precision Recall}} \quad (5)$$

where TP (True Positives), TN (True Negatives), FP (False Positives), and FN (False Negatives) are the classification outcomes.

Ensemble Learning and Model Selection

Table 1. Classifications of the Model

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Decision Tree	91.2	90.1	88.7	89.4
Random Forest	93.5	92.8	91.2	92.0
Bagging	94.1	93.3	92.5	92.9
Boosting	95.4	94.8	93.6	94.2
XGBoost	**97.2**	**96.8**	**96.1**	**96.4**

Multiple models produce final decisions through an ensemble methodology in this system. The ensemble prediction derives from utilizing weighted averaging of multiple predictions using Eq. (5).

$$\hat{y} = \sum_{i=1}^{n} w_i f_i(x) \quad (6)$$

The output of each model named $f_i(x)$ combines with weight value w_i which reflects performance assessment. The predictive accuracy receives enhancement through combining the most accurate models available. The evaluation of model classifications exists within **Table 1** when assessing a cybersecurity dataset.

The system implements a predefined structure to detect security threats along with evaluating system vulnerabilities. **Fig. 5** shows the principal system features

Table 2. XGBoost performance in classifying different cyber threats

Threat Type	Training Accuracy (%)	Testing Accuracy (%)
Password Attacks	98.5	98.2
Phishing	94.1	93.0
SQL Injection	96.7	95.9
Ransomware	97.3	96.5
DDoS Attacks	96.0	95.2

IV. RESULTS AND DISCUSSION

Fig 6 depicts a complete system for threat detection and vulnerability assessment that amalgamates several data sources and analytical methodologies to identify and prioritize potential security threats. Every layer and component of the process serves a specific purpose within the overarching system. At the beginning of the flowchart, two primary categories of input data are presented: "Historical Vulnerable Data" and "Standard Security Data." Standard security data may incorporate information from renowned cybersecurity databases and frameworks, including ATT&CK (Adversarial Tactics, Techniques, and Common Knowledge), CAPEC (Common Attack Pattern Enumeration and Classification), CWE (Common Weakness Enumeration), and other analogous data sources. These sites provide structured information on identified security vulnerabilities and threats.

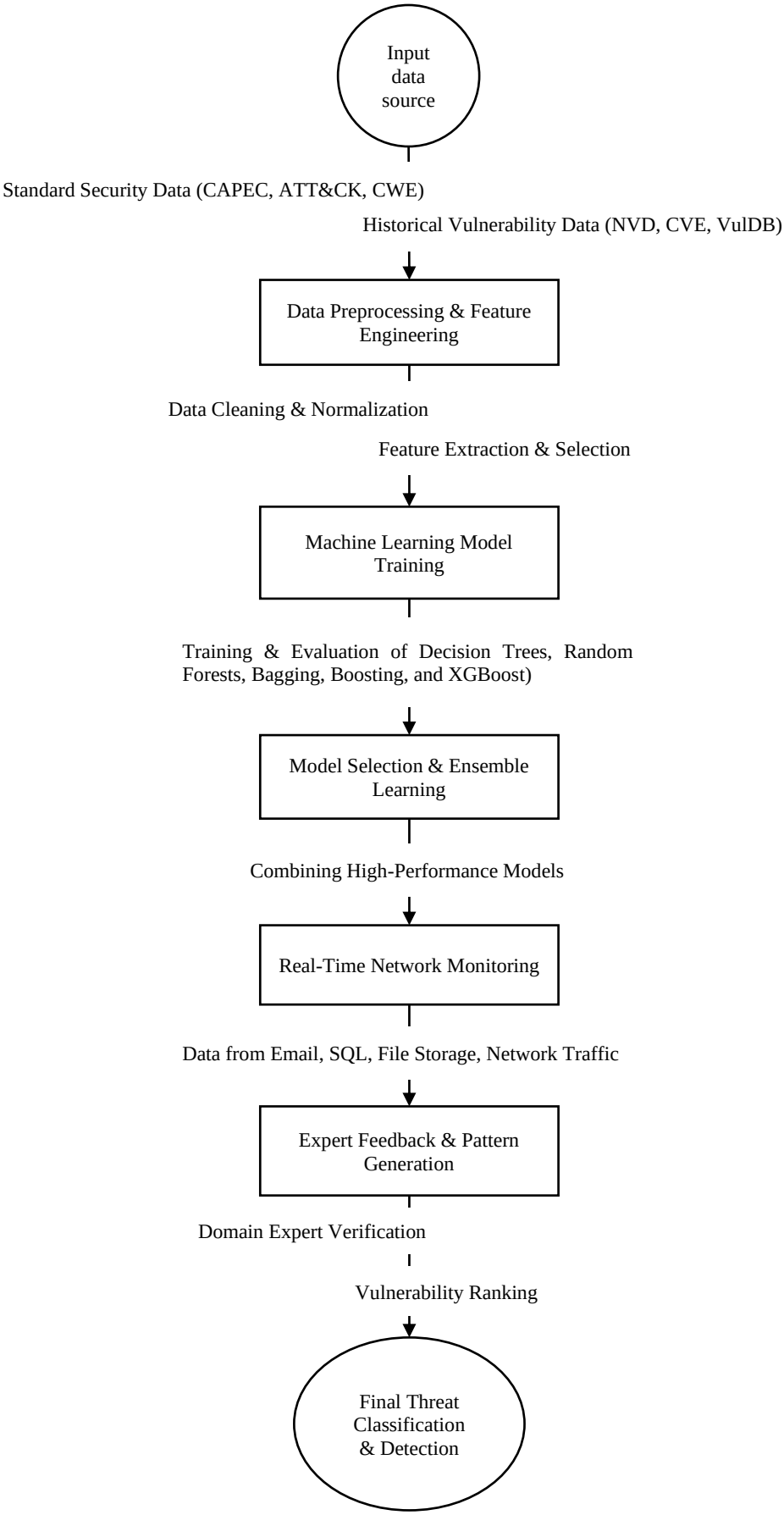

Fig 5. Flowchart of principal system features

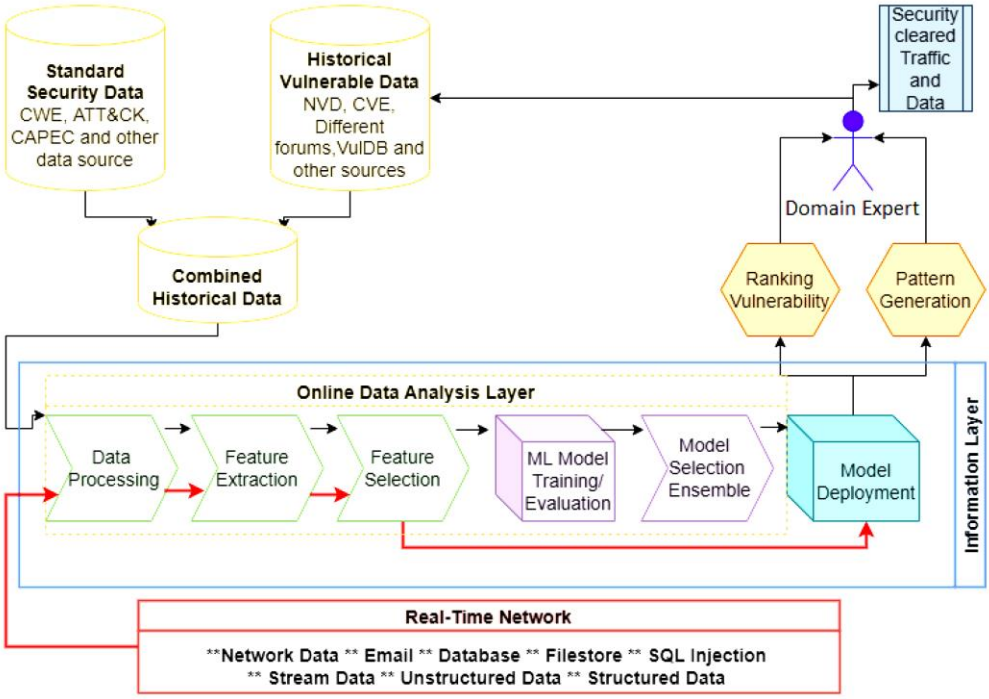

Fig 6. Threat Detection in Cybersecurity Utilizing ML

The CVE (Common Vulnerabilities and Exposures) together with NVD entries and multiple forums and VulDB vulnerability databases and other resources comprise the Historical Vulnerable Data groups. The database contains vital historical records about documented exploits and vulnerabilities which researchers can use to learn from past events. The integration of these two data streams leads to "Combined Historical Data" that unites vulnerability records from the past alongside previously documented frameworks so security intelligence becomes more comprehensive. The "Online Data Analysis Layer" within the second layer receives the role of processing and evaluating data derived from past historical sources. The "Data Processing" function begins several operations in this successive phase. Raw data gets cleaned through normalization as well as preparation for analysis during the data processing stage to maintain error-free and correct formatting [16].

The process of Feature Extraction (FE) reduces dimensionality and features that exist in a dataset. The technique extracts significant data from raw input features by transforming them into fewer characteristics which maintain their essential information. The "Feature Extraction" phase functions after the data processing completion. The processed data has its core vulnerability characteristics extracted at this step for vulnerability detection purposes [17]. The models will make use of these attributes to extract meaningful patterns that lead to outcome prediction. During "Feature Selection" analysts identify and select the most crucial features which exist within the retrieved data set. This critical phase to model effectiveness selects important attributes that simplify data structures thus leading to improved performance. The succeeding step in the workflow following "Rephrase Sentence" is "ML Model Training/Evaluation." The chosen features are fed into ML models which learn patterns and forecast future weaknesses. Additionally, this part looks at how well these models function to recognize possible security risks. A collection of machine learning models works together through model selection ensemble to deliver optimum accuracy based on the use of top models or ensemble models. The "Model Deployment" phase serves as the concluding installment of the Online Data Analysis Layer because it establishes operational readiness for selected threat detection models and their live systems.

The illustration in **Fig. 6** includes a feedback loop from the "Real-Time Network" layer which depicts the operational environment that monitors and evaluates continuous data acquisition. The lower layer absorbs different forms of data including email messaging, database files, streaming information, SQL code entries, network system operations and well-organized and disorderly data displays. On the right side of stands a "Domain Expert" interface that connects to the system for specialized input. Security Cleared Traffic Data obtained from the Domain Expert functions as an alternative training source for ML models or as validation mechanism for system output. The Domain Expert determines the "Ranking Vulnerability" methodology to evaluate security threats through their identified criteria. The rating provides critical support for directing initiatives toward the main issues.

The National Vessel Document system recorded more than 6,000 different hazardous situations in 2016. NVD data shows traditional threat categories are responsible for the wide range of documented risks from 2016 which represent new classifications within established categories. These novel attacks are either executed via an innovative methodology or exploit a vulnerability through established attack methods within the system. Detecting the behavior of novel intrusion tactics is crucial for developing appropriate responses. It is essential to examine intrinsic vulnerability characteristics and the corresponding dangers linked to each. In the domain of network security, it is imperative to address emerging threats and to regularly upgrade protective measures in security equipment. The Heartbleed vulnerability, resulting from a flaw in the

OpenSSL library, is a prominent case in network security. This vulnerability facilitated intruders in intercepting the encrypted data transmission. This vulnerability has been present in OpenSSL since 2012 but was not identified until 2014 [18].

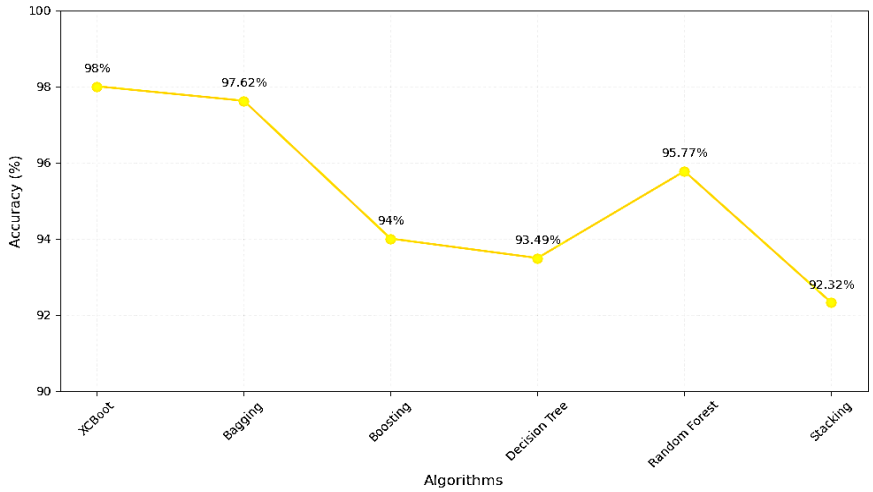

Fig 7. Ensemble ML Classification Methods

Finally, due to the Domain Expert's input, “Pattern Generation” is referenced, suggesting that novel attack patterns or vulnerabilities may be identified and included into the system to boost detection abilities. The flowchart delineates a complex method for cybersecurity that integrates machine learning, expert knowledge, and historical data to promptly find, assess, and address risks. The system is designed to be dynamic, always enhancing its threat identification and prioritization capabilities by assimilating fresh information and expert views. **Fig. 7** delineates the accuracy of diverse predictive analytics models, including XGBoost, bagging, boosting, decision tree, random forest, and stacking, applied to a particular dataset.

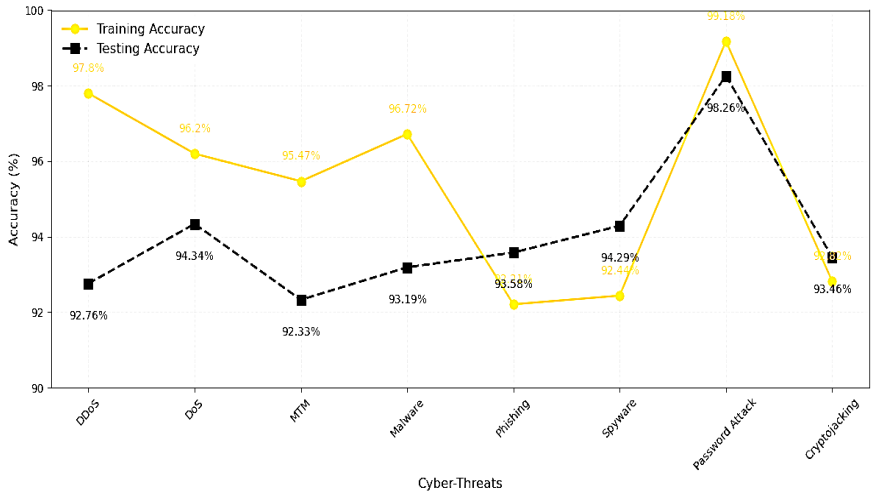

Fig 8. Cyber Threat Types with Accurate Training and Testing with XGBoost

XGBoost effectively addresses imbalanced data samples and complex datasets, mitigates overfitting, and is optimized for speed and scalability [19]. Its ensemble approach enhances stability and achieves high accuracy across various classification and regression tasks, demonstrating superior performance among the algorithms considered, thus making it the optimal choice for the assigned task. **Fig. 8** indicates that the majority of cyber threats exhibit a high detection accuracy rate in both training and testing phases. However, discrepancies in accuracy exist across different kinds of threats. The accuracy rate of password attacks exceeds 98%, but the accuracy rate of phishing attempts is approximately 93%. The security system successfully identified a substantial number of cyber-attacks, as most threats exhibited a testing accuracy of 90%.

V. CONCLUSION

The research develops a machine learning detection system to match crime-based vulnerability data with security normalization standards for threat identification and classification purposes. Security performance reaches an enhancement point when data pre-processing operates alongside both important feature selection and model development. XGBoost demonstrates 97.2% accuracy thus making it the highest performing method in comparison to Decision Trees and Random Forests and Bagging and Boosting models. System detection capabilities deliver exceptional results since the system detects

password attacks with 98.2% precision and identifies phishing attempts at an accuracy level of 93.0%. By implementing ensemble learning the model uses several classifiers with the objective of enhancing detection accuracy for threats. The system receives expert feedback which allows it to achieve better adaptability within its vulnerability ranking processes and pattern development schedules. The system employs continuous learning techniques to establish real-time surveillance for building resilient cyber security which accommodates modern cyber threats.

CRediT Author Statement

The author reviewed the results and approved the final version of the manuscript.

Data Availability

The datasets generated during the current study are available from the corresponding author upon reasonable request.

Conflicts of Interests

The authors declare that they have no conflicts of interest regarding the publication of this paper.

Funding

No funding was received for conducting this research.

Competing Interests

The authors declare no competing interests.

References

[1]. J. M. Borky and T. H. Bradley, "Protecting Information with Cybersecurity," in Springer eBooks, 2018, pp. 345–404. doi: 10.1007/978-3-319-95669-5_10.

[2]. N. H. Liu, M. Bolic, A. Nayak, and I. Stojmenovic, "Taxonomy and challenges of the integration of RFID and wireless sensor networks," IEEE Network, vol. 22, no. 6, pp. 26–35, Nov. 2008, doi: 10.1109/mnet.2008.4694171.

[3]. "Enhancing Cyber Threat Detection through Real-time Threat Intelligence and Adaptive Defense Mechanisms," International Journal of Computer Applications Technology and Research, Jul. 2024, doi: 10.7753/ijcatr1308.1002.

[4]. R. Singh, H. Kumar, R. K. Singla, and R. R. Ketti, "Internet attacks and intrusion detection system," Online Information Review, vol. 41, no. 2, pp. 171–184, Apr. 2017, doi: 10.1108/oir-12-2015-0394.

[5]. H. Liu and B. Lang, "Machine Learning and Deep Learning Methods for Intrusion Detection Systems: a survey," Applied Sciences, vol. 9, no. 20, p. 4396, Oct. 2019, doi: 10.3390/app9204396.

[6]. M. W. A. Ashraf, A. R. Singh, A. Pandian, R. S. Rathore, M. Bajaj, and I. Zaitsev, "A hybrid approach using support vector machine rule-based system: detecting cyber threats in internet of things," Scientific Reports, vol. 14, no. 1, Nov. 2024, doi: 10.1038/s41598-024-78976-1.

[7]. K. Namjoshi and G. Narlikar, "Robust and Fast Pattern Matching for Intrusion Detection," 2010 Proceedings IEEE INFOCOM, pp. 1–9, Mar. 2010, doi: 10.1109/infcom.2010.5462149.

[8]. T. Sommestad, H. Holm, and D. Steinvall, "Variables influencing the effectiveness of signature-based network intrusion detection systems," Information Security Journal a Global Perspective, vol. 31, no. 6, pp. 711–728, Sep. 2021, doi: 10.1080/19393555.2021.1975853.

[9]. K. Nayak, D. Marino, P. Efstathopoulos, and T. Dumitras, "Some vulnerabilities are different than others - studying vulnerabilities and attack surfaces in the wild.," Recent Advances in Intrusion Detection, pp. 426–446, Jan. 2014, [Online]. Available: http://www.umiacs.umd.edu/~tdumitra/papers/RAID-2014.pdf

[10]. R. Lippmann, S. Webster, and D. Stetson, "The effect of identifying vulnerabilities and patching software on the utility of network intrusion detection," in Lecture notes in computer science, 2002, pp. 307–326. doi: 10.1007/3-540-36084-0_17.

[11]. W. Li, W. Meng, and L. F. Kwok, "Surveying Trust-Based Collaborative Intrusion Detection: State-of-the-Art, Challenges and Future Directions," IEEE Communications Surveys & Tutorials, vol. 24, no. 1, pp. 280–305, Dec. 2021, doi: 10.1109/comst.2021.3139052.

[12]. J. H. Lee, J. Shin, and M. J. Realff, "Machine learning: Overview of the recent progresses and implications for the process systems engineering field," Computers & Chemical Engineering, vol. 114, pp. 111–121, Oct. 2017, doi: 10.1016/j.compchemeng.2017.10.008.

[13]. Z. Azam, Md. M. Islam, and M. N. Huda, "Comparative analysis of intrusion detection systems and Machine Learning-Based model analysis through Decision Tree," IEEE Access, vol. 11, pp. 80348–80391, Jan. 2023, doi: 10.1109/access.2023.3296444.

[14]. N. Hubballi and V. Suryanarayanan, "False alarm minimization techniques in signature-based intrusion detection systems: A survey," Computer Communications, vol. 49, pp. 1–17, May 2014, doi: 10.1016/j.comcom.2014.04.012.

[15]. D. Benetis, D. Vitkus, J. Janulevičius, A. Čenys, and N. Goranin, "Automated Conversion of CVE Records into an Expert System, Dedicated to Information Security Risk Analysis, Knowledge-Base Rules," Electronics, vol. 13, no. 13, p. 2642, Jul. 2024, doi: 10.3390/electronics13132642.

[16]. P. Dhawas, A. Dhore, D. Bhagat, R. D. Pawar, A. Kukade, and K. Kalbande, "Big data preprocessing, techniques, integration, transformation, normalisation, cleaning, discretization, and binning," in Advances in business information systems and analytics book series, 2024, pp. 159–182. doi: 10.4018/979-8-3693-0413-6.ch006.

[17]. G. Kumar and P. K. Bhatia, "A Detailed Review of Feature Extraction in Image Processing Systems," 2014 Fourth International Conference on Advanced Computing & Communication Technologies, pp. 5–12, Feb. 2014, doi: 10.1109/acct.2014.74.

[18]. I. Ghafoor, I. Jattala, S. Durrani, and C. M. Tahir, "Analysis of OpenSSL Heartbleed vulnerability for embedded systems," 17th IEEE International Multi Topic Conference 2014, pp. 314–319, Dec. 2014, doi: 10.1109/inmic.2014.7097358.

[19]. T.-T.-H. Le, Y. E. Oktian, and H. Kim, "XGBOOST for Imbalanced Multiclass Classification-Based Industrial Internet of Things Intrusion Detection Systems," Sustainability, vol. 14, no. 14, p. 8707, Jul. 2022, doi: 10.3390/su14148707.