Defensive Strategy for Epidemic Cyber Security threat Modelling and Analysis Using Support Vector Machines

S. Kasthuri[1], Dr. M. Balamurugan[2], A. Loganayaki[3]
[1]*PG Student, Department of Computer Science and Engineering, The Kavery Engineering College (Autonomous), Mecheri, Salem, Tamil Nadu, India–636 453*
[2]*Associate Professor and HOD, Department of Computer Science and Engineering, The Kavery Engineering College (Autonomous), Mecheri, Salem, Tamil Nadu, India–636 453*
[3]*Assistant Professor, Department of Computer Science and Engineering, The Kavery Engineering College (Autonomous), Mecheri, Salem, Tamil Nadu, India–636 453*

Abstract **- Digital systems, networks, and data require vital domain protection against malicious intrusion and unauthorized access through cybersecurity measures. Epidemic cyber threats like rapid malware and ransomware, and botnets threaten the domain because they exploit system vulnerabilities while spreading between connected networks. Today's Security threat models depend mainly on signature-based methods with heuristic detection methods, yet struggle to detect new attacks because they require defined rule sets and attack signatures. The traditional methods experience both computational inefficiency and numerous incorrect identifications that reduce their capability to stop security threats when they occur in real time. A defensive strategy for epidemic cyber threat modelling and analysis with Support Vector Machines (SVM) serves as the main proposal in this paper to address existing challenges. The model starts with feature value normalization through Min-Max scaling before model convergence optimization. The next step incorporates Recursive Feature Elimination (RFE) to choose essential features that boost classification results and diminish computing requirements. We employ SVM for classification because this method works well with large datasets while properly identifying benign from malicious computer operations. Our proposed method shows experimental result of increasing threat detection accuracy through built-in false positive rate which establishes it as a strong solution for contemporary cybersecurity protection.**

Keywords: cybersecurity, malicious, threat, Min-max scaling, RFE, SVM, benign

I. INTRODUCTION

The security measures against epidemic cyber threats have gained absolute importance in the evolving cybersecurity landscape. The fundamental element of cybersecurity threat analysis and modeling helps identify digital system vulnerabilities because it enables the protection of critical data networks, together with national infrastructure. Future security needs demand that companies develop proactive defense systems that protect their information from breaches and stop malware attacks and numerous cyber assaults. Historically, cybersecurity designs depended on detection rules, signature-based intrusion prevention tools, and heuristic scanners to spot cyber-attacks [1]. The standard security measures show limited protection against new and complex cyber-attacks, including zero-day exploits, Advanced Persistent Threats (APTs), and botnet-driven intrusions, because they only recognize pre-established attack patterns.

The main weakness of conventional cybersecurity strategies is the lack of dynamic response capabilities toward developing cyber threats. Both Signature-based Intrusion Detection Systems (IDS) and rule-based firewalls depend on regular maintenance and continuous updates. Yet, identifying new threats could face time delays [2]. The capacity of heuristic anomaly detection to identify dangerous actions comes with a significant drawback because it produces excessive false positives, which result in costly inefficiencies in real-time threat mitigation [3]. Advanced security mechanisms must be developed to address the growing sophistication of cyber-attacks because cybercriminals now use Artificial Intelligence (AI)-based threats, botnets, and polymorphic viruses. Detection accuracy and manual threat analysis reduction require automated decision-making and data processing

capabilities because of the excessive network traffic and cyber threat data volumes.

The proposed solution builds an epidemic cyber threat modeling system by implementing Support Vector Machines (SVM) for machine learning protection against attacks. The managed learning algorithm, SVM, is a potent solution because it produces excellent outcomes in diverse classification applications while performing strong anomaly detection. Through statistical learning, SVM-based cybersecurity solutions measure extremely voluminous network traffic and system logs to identify precise patterns of cyber threats [4]. SVM proves especially effective for cybersecurity applications because it addresses the specific need of working with imbalanced datasets which occur frequently in cyber threat detection when threat activities remain minimal compared to network traffic volume. Implementing SVM within cybersecurity defense systems creates a detection system that boosts performance while eliminating false alarms and enables fast identification of present and new cyberattacks. The presented work contributes to developing adaptive machine learning-based cybersecurity frameworks that adapt to emerging threats in the evolving cyber threat domain [5].

II. LITERATURE SURVEY

The author [6] created an Optimized Machine Learning Framework (OMLF) that utilizes feature selection techniques and hyperparameter adjustment to boost detection precision. The system struggles with device-independent attacks because it requires pre-labeled datasets for operation. The issue gets resolved through the implementation of Deep Reinforcement Learning (DRL) systems that adaptively learn threat patterns for better real-time cybersecurity threat detection and protection according to [7]. When deployed, deep reinforcement agents present difficult barriers because they demand costly and complex training procedures.

The intelligent phishing detection system the author created uses Convolutional Neural Networks and Recurrent Neural Networks (CNN-RNN). Still, it faces scalability problems and limitations in zero-day phishing attack detection. [9] established a supervised Machine Learning (ML)- based insider threat detection system, employing Decision Trees (DT) and Random Forests (DT-RF) to analyze user behavior patterns. The system produces many incorrect positives, which generate incorrect alerts for system administrators.

The author [10] developed a phishing attack detection system that uses SVM with K-Nearest Neighbors (SVM-KNN). Feature engineering affects the system's performance, reducing its ability to detect recently emerging phishing techniques during real-time operations. The authors in [11] developed a Deep Embedded Neural Network Expert System (DeNNeS), which unites deep autoencoders with classification networks to detect cyberattacks. The system needs numerous labeled data collections to function effectively, but dynamic cyber environments present challenges for its practical application.

The author [12] designed a system to detect malicious URLs by creating classification models using Logistic Regression and Gradient Boosting (LR-GB). This technique restricts the ability to detect malicious URLs because it relies on human-generated features, which makes it inadequate for encrypting malicious Internet addresses. A deep learning-based [13] proposed an intelligent phishing detection mechanism with Bi-LSTM (Bidirectional LSTM) and GRU (Gated Recurrent Units) models for threat analysis. However, its scalability suffers from the need for extensive labeled dataset requirements.

The author developed A spam detection method using Artificial Neural Networks (ANN) and sine-cosine algorithm features for selection. The enhanced classification results from this method come at the cost of longer optimization processing time. The [15] established a software-defined network-based phishing prevention system that incorporated the CANTINA approach supported by deep machine learning with Deep Belief Networks (DBN) to achieve enhanced real-time phishing detection. Such implementations need extensive infrastructure modifications, creating both practical and technical challenges for deploying this method in actual settings.

III. PROPOSED METHODOLOGY

This section briefly describes the proposed method for analysing cybersecurity threats. The NSL-KDD dataset is used to perform the proposed method, which

involves three processes: preprocessing, feature selection, and classification. The Min-max scaling method is used for preprocessing the NSL-KDD dataset, RFE for feature selection, and SVM for classifying normal or attack.

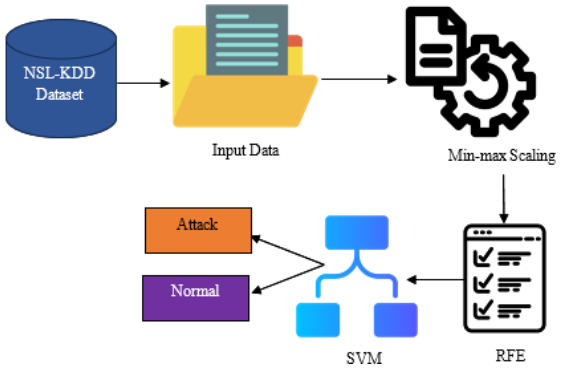

Figure 1. Architecture diagram of the proposed method

The proposed cybersecurity threat analysis method has its architectural diagram in Figure 1. The Min-max scaling method operates first to normalize the dataset through preprocessing because it brings consistency while reducing data variability. RFE operates in the second phase by choosing the crucial attributes that boost classification accuracy while dropping unneeded attributes or features with lower significance. The processed data moves to an SVM classifier which performs effective attack vs normal traffic detection.

A. Min-max Scaling

The min-max scaling normalization strategy converts the NSL-KDD dataset features into the range 0 to 1. The data distribution remains intact after min-max scaling because the methodology places values between 0 and 1 and improves model performance for algorithms that depend on feature magnitude values. Through min-max scaling, anomaly detectors can detect the NSL-KDD network traffic features with different numerical ranges because it ensures consistent and reliable training model performance.

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

The original network traffic feature value X appears is combined with that feature's minimum X_{min} and maximum X_{max} values within the dataset. The transformation method rescales all features into [0,1] without distorting relative value relationships and removes magnitude variations. The heterogeneous packet-driven features in NSL-KDD require min-max normalization to normalize values, increasing the model accuracy and efficiency for intrusion detection. The generalized version of min-max scaling allows specific range rescaling which works in [-1,1],

$$X' = a + \frac{(X - X_{min})(b - a)}{X_{max} - X_{min}} \quad (2)$$

The equation features a and b to establish target normalization ranges, which offer adaptable feature standardization options. The scaling technique provides an advantage to ML algorithms in the NSL-KDD dataset because these models work best with symmetric data distributions, including deep learning models. The process transforms network attributes into uniform scales, maintaining effective processing of numerical features. Categorical features are turned into numerical values without limits caused by feature magnitude differences.

B. Recursive Feature Elimination (RFE)

The systematic approach of Recursive Feature Elimination (RFE) is crucial in identifying essential features after min-max scaling preprocessing of the NSL-KDD dataset for intrusion detection. RFE works through ML model training while it ranks features by their importance, followed by stepwise feature removal until finding the most suitable subset of features. The feature selection process starts with all initial features, while an estimator, like SVM, assigns weight values to features for ranking purposes. A single feature receives the lowest evaluation from the model performance, which is removed during the model retraining process. The algorithm maintains the feature elimination procedure until it reaches the targeted feature count. RFE achieves efficiency and accuracy improvement through its ability to lower dimensions for better model generalization by reducing overfitting problems. RFE operates on NSL-KDD traffic data for feature selection to identify discriminative network traffic attributes, thus improving intrusion detection because it focuses on only essential features.

$$I_f = \sum_{i=1}^{n} w_i^2 \quad (3)$$

The I_f score for each feature results from summing the squared w_i weight coefficients the ML model assigned to all features. The starting point involves training a model using the entire selection of features after which the model assigns weights to determine feature importance for prediction. Model decisions are heavily affected by features carrying greater weight values. Network traffic attributes within the NSL-KDD dataset obtain different importance scores through weight coefficient squared calculations to determine relevant features for intrusion detection. The weighted equation preserves balanced influence between positive and negative model performance indicators by applying the weight-squaring method.

$$f_r = \arg\min_f I_f \quad (4)$$

The elimination process identifies the least necessary feature f_r according to its smallest significance score I_f. The model removes the feature that offers minimal contribution to its decision-making process following the first ranking operation. The recursive elimination method supports dimension reduction of the NSL-KDD dataset by keeping essential network traffic features active for anomaly detection. The removal system keeps unimportant features from affecting model performance by permitting only critical attributes to remain.

$$M_{k+1} = Train(M_k, F - f_r) \quad (5)$$

Removing the least essential feature enables a new training process for the ML model using its updated feature collection. The novel model referred to as M_k is the starting point, yet the subsequent retraining results in M_{k+1}. The feature set F undergoes an update through f_r removal because this modification helps the model adjust its learning parameters effectively. The NSL-KDD dataset benefits from this step because it helps optimize intrusion detection models through a reduced feature set, which maintains higher information content. The iteration-based retraining process operates until it retains only the most essential features.

$$|F| = d \quad (6)$$

The termination point of the feature elimination process happens when the threshold d is satisfied by the number of retained features $|F|$. RFE stops operations when the specified threshold d is reached to maintain sufficient data information in the NSL-KDD dataset classification process. When selecting a proper d value, the model achieves an optimal balance between data simplification and prediction accuracy. When the termination condition is satisfied, the training process employs the final collection of features to develop the intrusion detection model, which delivers better efficiency and detection outcomes.

C. Support Vector Machine (SVM)

The Support Vector Machine (SVM) is a widespread ML algorithm that classifies network traffic within the NSL-KDD dataset after RFE executes feature selection. SVM serves as a supervised learning method that establishes the best hyperplane for separating different classes present in the feature space. The selected features receive transfer into a higher-dimensional context that enables clear decision boundaries to form. SVM successfully identifies between normal and attack network traffic while employing the top features that the RFE algorithm previously chose from the NSL-KDD dataset. The technology proves successful for intrusion detection systems because it operates on high-dimensional data and shows substantial accuracy during classification procedures.

$$f(x) = w^T X + b \quad (7)$$

The SVM decision function uses X feature vectors to determine their classification group. Each selected feature obtains its weight value from the w weight vector while b defines the boundary limitations. The amount of similarity between a feature vector X and the defined hyperplane gets calculated through the dot product $w^T X$. The decision function returns $f(x)$ that reveals an input X as being part of one class for positive or greater-than-zero values (normal traffic in NSL-KDD) or identifying it as part of the second class (attack traffic) when the value is negative. The classification rule enables network intrusion detection to rely on the most crucial features that RFE selects.

$$M = \frac{2}{\|w\|} \quad (8)$$

In SVM, the margin M represents the distance between the closest data points (support vectors) belonging to separate classes. Better model generalization occurs with larger margins, reducing overfitting risks. When $\|w\|$ increases, the measurement of parallel distance (margin) reduces proportionally. SVM achieves robust attack and normal traffic pattern classification of the NSL-KDD dataset by maximizing this margin while minimizing classification errors.

$$\min_{w,b} \frac{1}{2}\|w\|^2 \quad (9)$$

subject to:

$$y_i(W^T X_i + b) \geq 1, \ \forall i \quad (10)$$

SVM operates to determine the ideal w and b values, which define the best possible decision boundary. The function optimizations minimize half the squared value of $\frac{1}{2}\|w\|^2$ to determine the margin width directly. The $y_i(W^T X_i + b) \geq 1$ stands as an essential condition for training each data point X_i to obtain proper classification results when y_i represents the class labels (+1 for normal and -1 for attack categories). The solution of this constrained optimization problem enables SVM to select the most suitable hyperplane whose function identifies normal versus attack traffic patterns within the NSL-KDD dataset.

$$K(X_i, X_j) = \emptyset(X_i)^T \emptyset(X_j) \quad (11)$$

The NSL-KDD dataset features require non-linear separation, which SVM performs through a kernel function $K(X_i, X_j)$ to handle the transformation into a higher-dimensional space for linear classification. Through mapping by function $\phi(X)$ the initial feature vector transforms into a new space where it becomes easier to classify. Different kernel functions enable SVM to handle complicated patterns using the Linear Kernel $\left(K(X_i, X_j) = X_i^T X_j\right)$ for linearly separable data. Polynomial Kernel $\left(K(X_i, X_j) = \left(X_i^T X_j + c\right)^d\right)$ captures non-linear relationships through polynomial transformations. By using Radial Basis Function (RBF) Kernels $\left(K(X_i, X_j) = \exp\left(-\gamma\|X_i - X_j\|^2\right)\right)$, data gets mapped to an infinite number of dimensions, which enables the classification of highly complex boundaries. The application of kernel functions in SVM enables the selection of crucial features from the NSL-KDD dataset, enhancing the efficiency of intrusion detection.

IV. RESULT AND DISCUSSION

This section evaluates the proposed method's performance using accuracy, precision, recall, F1 score, and time complexity. This evaluation determines a reliable and accurate method for analyzing cybersecurity threats using the deployed method with DRL, DT-RF, and ANN.

Table 1. Simulation Parameters

Parameters	Values
Dataset Name	NSL-KDD
Training Records	125,973
Test Records	22,544
No of Features	41
Language	Python
Tool	Jupyter Notebook

Table 2 shows the simulations and parameters of the paper. The NSL-KDD dataset is used to classify cybersecurity threats, contains 125,973 training records, and 22,544 test records. The implementation method is run using Python and the Jupyter Notebook Tool.

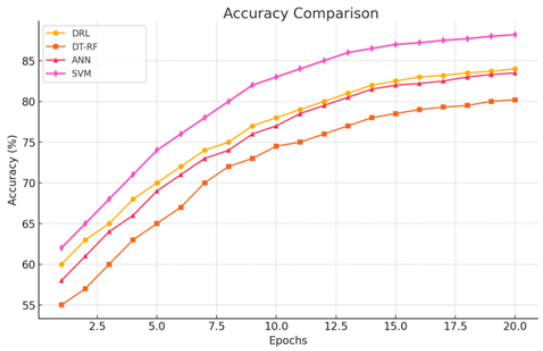

Figure 2. Analysis of accuracy performance

The analysis of network traffic patterns depends on several ML applications, including DRL, DT-RF and ANN, and SVM methods, as Figure 2 shows. The research SVM operates effectively as a proposed method to enhance classification accuracy and delivers better results than existing methods when using multiple training epochs. Researchers use this dataset as a vital basis to test and validate new advanced cybersecurity solutions.

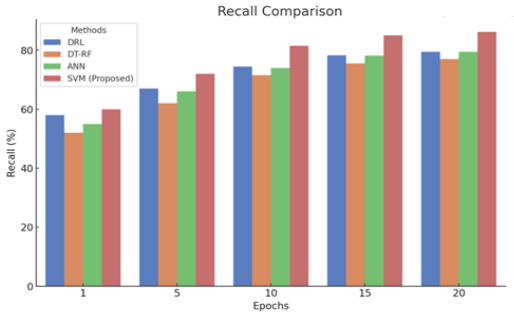

Figure 3. Analysis of recall performance

The classification of network intrusions uses multiple ML techniques, including DRL, DT-RF, ANN, and SVM, as shown in Figure 3. SVM is a proposed method for enhancing recall performance by achieving superior results during multiple epochs compared to present techniques. The NSL-KDD dataset is fundamental for creating and evaluating progressed cybersecurity solutions.

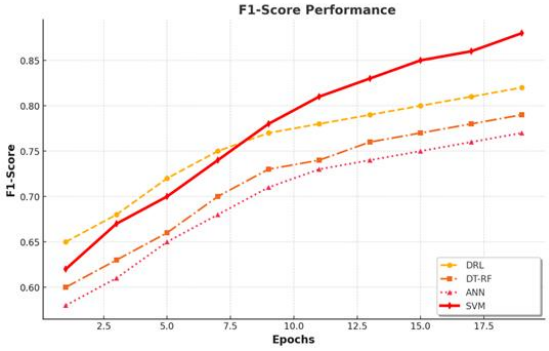

Figure 4. Analysis of F1-score performance

The performance evaluation of ML methods, DRL, DT-RF, ANN, and SVM, through Figure 4, utilizes NSL-KDD as its primary assessment tool. The dataset's design provides precise performance evaluation through validations of accuracy and F1-score, along with detection rate, which leads to the development of highly secure intrusion detection systems.

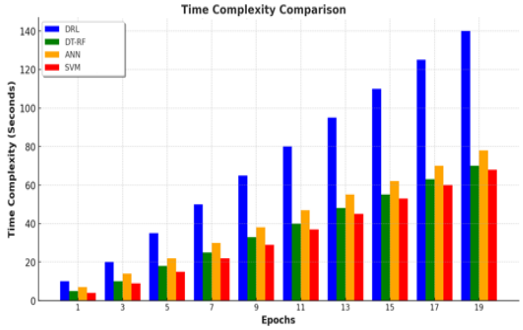

Figure 5. Analysis of time complexity performance

Figure 5 shows the effectiveness assessment for different machine learning algorithms, such as DRL, DT-RF, ANN, and SVM, using NSL-KDD. The dataset facilitates the creation and enhancement of effective intrusion detection models through its accuracy assessment, F1-score metrics, and time complexity measures throughout different training periods.

V. CONCLUSION

In summary, we developed a defensive methodology to model and analyze epidemic cyber threats by applying SVM on NSL-KDD dataset. The model first normalized features using Min-max scaling followed by performing RFE to extract important features that were subsequently classified by SVM. The method delivered improved attack detection capabilities together with fewer unneeded results in false positive outcomes. A powerful defence mechanism in cybersecurity becomes achievable through this approach which generates accurate classifications of threatening versus non-threatening activities. The analysis proves that ML enhances cybersecurity through its capacity to handle dynamic threats and that both feature refinement and data clearing lead to superior classification results.

REFERENCE

[1] Sarker, I. H., Abushark, Y. B., Alsolami, F., & Khan, A. I. (2020). IntruDTree: A Machine Learning Based Cyber Security Intrusion Detection Model. Symmetry, 12(5), 754. https://doi.org/10.3390/sym12050754

[2] Sarker, I. H. (2021). CyberLearning: Effectiveness analysis of machine learning security modeling to detect cyber-anomalies and multi-attacks. Internet of Things, 14, 100393. https://doi.org/10.1016/j.iot.2021.100393

[3] Srinivasan, S., & P, D. (2023). Enhancing the security in cyber-world by detecting the botnets using ensemble classification based machine learning. Measurement: Sensors, 25, 100624. https://doi.org/10.1016/j.measen.2022.100624

[4] Aiyanyo, I. D., Samuel, H., & Lim, H. (2019). A Systematic Review of Defensive and Offensive Cybersecurity with Machine Learning. Applied Sciences, 10(17), 5811. https://doi.org/10.3390/app10175811

[5] Giovanni Apruzzese, Pavel Laskov, Edgardo Montes de Oca, Wissam Mallouli, Luis Brdalo Rapa, Athanasios Vasileios Grammatopoulos, and Fabio Di Franco. 2023. The Role of Machine Learning in Cybersecurity. Digital Threats 4, 1, Article 8 (March 2023), 38 pages. https://doi.org/10.1145/3545574

[6] Omer, N., Samak, A. H., Taloba, A. I., & Abd El-Aziz, R. M. (2024). Cybersecurity Threats Detection Using Optimized Machine Learning Frameworks. Computer Systems Science & Engineering, 48(1).

[7] Sewak, M., Sahay, S.K. & Rathore, H. Deep Reinforcement Learning in the Advanced Cybersecurity Threat Detection and Protection. Inf Syst Front 25, 589–611 (2023). https://doi.org/10.1007/s10796-022-10333-x

[8] Mughaid, A., AlZu'bi, S., Hnaif, A. et al. An intelligent cyber security phishing detection system using deep learning techniques. Cluster Comput 25, 3819–3828 (2022). https://doi.org/10.1007/s10586-022-03604-4

[9] Manoharan, P., Yin, J., Wang, H. et al. Insider threat detection using supervised machine learning algorithms. Telecommun Syst 87, 899–915 (2024). https://doi.org/10.1007/s11235-023-01085-3

[10] M. N. Alam, D. Sarma, F. F. Lima, I. Saha, R. -E. -. Ulfath and S. Hossain, "Phishing Attacks Detection using Machine Learning Approach," 2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT), Tirunelveli, India, 2020, pp. 1173-1179, doi: 10.1109/ICSSIT48917.2020.9214225.

[11] Mahdavifar, S., Ghorbani, A.A. DeNNeS: deep embedded neural network expert system for detecting cyber attacks. Neural Comput & Applic 32, 14753–14780 (2020). https://doi.org/10.1007/s00521-020-04830-w

[12] Do Xuan, C., Nguyen, H. D., & Tisenko, V. N. (2020). Malicious URL detection based on machine learning. International Journal of Advanced Computer Science and Applications, 11(1).

[13] Adebowale, M.A., Lwin, K.T. and Hossain, M.A. (2023), "Intelligent phishing detection scheme using deep learning algorithms", Journal of Enterprise Information Management, Vol. 36 No. 3, pp. 747-766. https://doi.org/10.1108/JEIM-01-2020-0036

[14] Talaei Pashiri, R., Rostami, Y. & Mahrami, M. Spam detection through feature selection using artificial neural network and sine–cosine algorithm. Math Sci 14, 193–199 (2020). https://doi.org/10.1007/s40096-020-00327-8

[15] S., E. R., & Ravi, R. (2020). A performance analysis of Software Defined Network based prevention on phishing attack in cyberspace using a deep machine learning with CANTINA approach (DMLCA). Computer Communications, 153, 375-381.https://doi.org/10.1016/j.comcom.2019.11.047