Volume (1) No (11), 2025
Journal of Global Research in Multidisciplinary Studies (JGRMS)
Review Paper/Research Paper
Available online at https://saanvipublications.com/journals/index.php/jgrms/index

Optimization of Ensemble ML Models for Cybersecurity Analytics Using CICIDS 2017

Dr Chintal Kumar Patel
Associate Professor, CSE
Geetanjali Institute of Technical Studies
chintal.patel@gits.ac.in

***Abstract*— Intrusion detection is one of the most significant components of modern cybersecurity systems due to the increase in the complexity of cyberattacks. Conventional detection approaches can be inefficient in handling the scale, imbalance, and complexity of network traffic, resulting in lower accuracy and an increase in false positives. This paper offers a hybrid ensemble model for intrusion detection using the CICIDS 2017 dataset, which combines Random Forest (RF) and Light Gradient Boosting Machine (LGBM). The methodology includes intensive data preprocessing, such as stratified random sampling, elimination of irrelevant features, aggregation of attack classes, and balancing of the distributions of classes using the SMOTE Tomek technique. The most important predictors are chosen using Pearson correlation and mutual information. The hybrid ensemble model capitalizes on the strengths that RF and LGBM have in complement and thus it can generalize better, minimize variance, and be more resistant to noisy and skewed data. The experimental findings indicate that the proposed method successfully attains 99.90% accuracy (ACC), 99.80% precision (PRE), 99.92% recall (REC) and 99.90% F1-score (F1), and can be used in comparison with standard ML models, including SVM and Naive Bayes. The results confirm the efficiency of ensemble learning in practice of intrusion detection and thus provide a scalable, precise, and dependable method of cloud-based cybersecurity detection.**

Keywords—Intrusion Detection System (IDS), CICIDS2017 dataset, Machine Learning, Ensemble Learning, SMOTETomek, Cybersecurity Analytics.

I. INTRODUCTION

The rapid technology of digitization of industries has extended the cybersecurity threat landscape to a dynamic, multifaceted, and dynamic battlefield [1]. With the increased migration of the world to cloud systems and interconnected systems, cyber threats have become more complex, and a variety of individual hackers or organized cybercrime gangs and state-funded cybercriminals have banded together with a wide variety of reasons, including financial benefits, espionage, and political intrusion [2][3]. The growing nature highlights the urgent requirement of proactive defense systems, early warning and defensive cybersecurity models so as to pass through the internet credibility and secure the respectable character of the cloud facilities.

ML has become a revolutionary technology in cybersecurity analytics, able to scale, adapt to the variety, and unpredictability of current cyber threats [4][5][6][7]. Traditional rule-based intrusion detection systems have been rendered useless in the face of ever-changing attack methods and the enormous volume of network traffic data [8]. ML-based models, however, have the ability to learn complex threat signatures, identify latent correlations and identify anomalies in real time. Intrusion detection, malware classification, and phishing detection are just a few of the many areas where they have proven effective. However, their large-scale implementation in a cloud-based system is hampered by the problems of data imbalance, adversarial control, and poor interpretability.

Ensemble machine learning has been used to address these shortcomings, as it is increasingly popular in optimizing cybersecurity analytics [9][10]. Ensemble techniques Bagging, boosting, and stacking are all methods of ensemble techniques, which are used to group several classifiers together to increase the prediction ACC, decrease variance and bias, and also improve the model generalization. The bagging methods reduce overfitting because parallel training on diverse and random data sets, incremental training increases the weakness of the learners by reducing errors, and stacking combines several base learners to create a meta-classifier with enhanced decision-making abilities [11][12]. These techniques have shown impressive results within the Intrusion Detection Systems (IDS) by exploiting the strengths complementary nature of different models.

The Canadian Institute of Cybersecurity's CICIDS2017 dataset allows for the comparison of intrusion detection system models [13]. It summarizes realistic network traffic information in various types of attacks DoS, DDoS, port scans, and infiltration attempts making it effective in testing ensemble models within the cybersecurity analytics platform on clouds [14]. Although missing values and class imbalance are some of the challenges encountered, the optimization of ensemble ML methods on CICIDS2017 can substantially improve threat detection performance, scale, and robustness of network-defined systems based on the cloud.

A. Significance and Contribution

This increasing reliance on cloud and interconnected digital structures has dramatically increased the vulnerabilities to cybersecurity, and therefore, contemporary networks have become more vulnerable to large-scale and advanced attacks. The conventional rule-based IDS may not be able to handle the volume, speed and diversity of the current network traffic-based data, resulting in a lag in or inaccurate threat identification. ML models, despite their strength, are characterized by challenges like data imbalance, adversarial manipulation, and lack of interpretability, among others, that

prevent their use in real-world cloud environments. To counter these issues, there is an urgent necessity of sophisticated, extensible, and smart intrusion detection models that could effectively be used to identify the anomalies in dynamic network environments. With its capability to harness the capabilities of a number of classifiers, ensemble learning offers a viable path ahead in improving detection ACC, robustness, and generalization in cybersecurity analytics. The key contributions are:

- The CICIDS 2017 dataset has been pre-processed in a systematic manner consisting of stratified sampling, deletion of irrelevant features, and class combination to form a balanced and representative dataset.
- The SMOTE Tomek resampling algorithm is used to overcome the issue of class imbalance and the improvement of model fairness between attack traffic and normal traffic.
- Pearson correlation and mutual information are used in the dual stage feature selection method to eliminate redundancy and keep the most distinguishing features.
- The proposed hybrid model outperforms classic ML models such as SVM and Naïve Bayes on indicators such as ACC, PRE, REC, F1, and ROC-AUC, which are cautiously evaluated.

B. Justification and Novelty

The conventional intrusion detection approaches have some constraints, such as failure to deal with complex attacks, uneven wealth distribution and high dimensionality of traffic data. These issues motivate work. In the present research, the hybrid ensemble model is introduced, which incorporates the Random Forest and LightGBM to address these issues. This framework is able to combine the variance reduction of bagging with the bias reduction of boosting successfully. The novelty lies in its end-to-end design: a robust pre-processing pipeline with SMOTETomek balancing and mutual-information-based feature selection ensures cleaner and more informative inputs, while the hybrid model delivers superior ACC, robustness, and reduced false positives compared to single classifiers. These components work in tandem to provide nearly flawless results on the CICIDS2017 dataset, providing a scalable and deployment-ready answer for actual IDS.

C. Structure of the Paper

The study is organized in the following way Intrusion detection systems and cybersecurity analytics are covered in Section II. Included in Section III are the specifics of the methods and materials employed, such as the dataset and the preprocessing techniques. In Section IV, offer the experimental results of the suggested hybrid ensemble system for intrusion detection. The examination and proposals for further study and implementation are summarized in Section V.

II. LITERATURE REVIEW

The most up-to-date literature on ML, DL, and cybersecurity analytics, intrusion detection systems, and reviews is covered in this section. Table I summarizes the authors, methods, datasets, key findings, and limitations or directions for future work.

Gowda (2025) introduces an innovative method for identifying financial threats with a hybrid model that integrates Gated Recurrent Units (GRU) and Capsule Networks. The proposed approach utilizes the temporal sequence processing abilities of GRU to identify patterns in transaction data, while capsule networks improve feature extraction and classification ACC via dynamic routing techniques. The model is trained and assessed on an extensive financial dataset, attaining a remarkable ACC of 98.85% [15].

Fu (2025) study shows the infrastructure of the system is composed of five levels: data gathering, data preparation, feature extraction, IDS, and warning and reaction. With this design, they may rest assured that the system is straightforward to manage, scalable, and efficient. The research found that the hybrid model, which combined DNN and RF, achieved a far better ACC rate of 98.5% on the test set compared to models that solely used one or the other. By categorizing network data in milliseconds, the technique meets the real-time needs of IDS [16].

Ogundokun, Owolawi and Van Wyk (2025) architecture integrates compact feature encoding with parallel ReLU- and Sigmoid-activated branches to enable strong multi-scale feature learning with low inference cost and simple architecture. Experimental results show that LiteRT-IDSNet achieved training ACC of 99.61% and validation ACC of 99.60%, outperforming the baseline model with 99.47% training and 99.59% validation ACC [17].

Zhang (2024) the features are extracted from pre-processed data by using Local Linear Embedding (LLE) which identified nonlinear structures and anomalies that indicated in the network intrusions. Finally, the network intrusions are detected by using LSTM which effectively improves ACC, reduces false positives, enhances the overall robustness in intrusion detection system. The proposed SCAE-LSTM achieved better ACC (0.9755), detection rate (0.9999), F1 value (0.9575) and FPR (0.0128) when compared with existing CNN-LSTM [18].

Khan et al. (2024) algorithms are pulled from literature studies because of their exceptional performance on old datasets. This work has achieved a DT model with 96.37% ACC and 96.33% F1 and the AdaBoost model with 96.37% ACC and 96.33% F1 for multiclass classification. Test ACC for binary classification has been 99.96% for the DT model, 99.84% for RF, 99.77% for Adaboost, and 99.57% for Xgboost; the top three models also had the best average PRE at 100% and ROC-AUC at 99.96% [19].

Filiz et al. (2024) of network traffic data obtained from a digital media and entertainment provider operating in Turkey is conducted through the application of multivariate time-series analysis techniques in order to get insights into the temporal patterns and trends. Models from LSTM and GRU have shown improved performance with respect to MATE and R-squared Score. At 8.95% MAPE, the LSTM model has achieved an R2 of 0.98 [20].

Sridevi et al. (2023) use DL and ML techniques to provide a thorough strategy for detecting insider threats. To identify unusual insider behaviors, and suggest a hybrid model that integrates DNN, which can grasp granular behavioral details with feature-engineered patterns. Unusual actions taken by insiders could be detected using this technique. The detection ACC of model was 96.3% [21].

Bokolo, Jinad and Liu (2023) A variety of malware applications' byte codes, section codes, and opcodes are combined and then classified using various techniques such as

RF, DT, SVM, KNN, SGD, LR, NB, and DL. The DL model has a high success rate of 96% thus showing better performance compared with other machine learning approaches [22].

Despite recent research that illustrates high ACC with hybrid and DL models in the detection of IDS and financial threats, a number of gaps still exist. The majority of the methods are based on particular datasets or models, such as GRU, LSTM, or capsule networks, and do not generalize well to various network conditions. Such challenges as class imbalance, high-dimensional feature redundancy, and real-time detection are not properly considered. Also, there is not much use of ensemble learning techniques which combine boosting and bagging to enhance robustness and generalization. Lastly, predictive explainability is a highly underestimated aspect, which limits its usage in cybersecurity processes. This shows the necessity of having scalable, interpretable and hybrid ensemble architecture of real-world intrusion detection.

TABLE I. SUMMARY OF BACKGROUND STUDY FOR MACHINE LEARNING AND DEEP LEARNING APPROACHES IN CYBERSECURITY AND INTRUSION DETECTION SYSTEMS

Author	Methods	Dataset	Key Findings	Limitations & Future Work
Gowda (2025)	Hybrid GRU + Capsule Networks	Large-scale financial dataset	Achieved 98.85% ACC; GRU captures temporal patterns while Capsule Networks enhance feature extraction via dynamic routing	Future work: real-time financial threat detection and scalability across institutions
Fu (2025)	Hybrid DNN + Random Forest in layered IDS architecture	Network traffic dataset	Achieved 98.5% ACC; real-time classification of traffic within milliseconds; system meets real-time IDS requirements	Explore large-scale deployment, scalability, and integration with heterogeneous IDS environments
Ogundokun, Owolawi & Van Wyk (2025)	LiteRT-IDSNet with compact feature encoding, parallel ReLU & Sigmoid branches	RT-IoT 2022 dataset (123,117 instances, 12 attack classes)	Training ACC 99.61%, validation 99.60%; outperformed baseline fully connected NN	Evaluate deployment on resource-constrained IoT devices and improve real-time detection efficiency
Zhang (2024)	SCAE + LSTM with Local Linear Embedding (LLE)	Network intrusion dataset	Achieved ACC 0.9755, F1-score 0.9575, low false positive rate (0.0128); superior to CNN-LSTM	Future work: reduce complexity and improve interpretability for large-scale deployment
Khan et al. (2024)	DT, RF, XGBoost, KNN, NB, LR, AdaBoost	Benchmark cybersecurity datasets	Binary classification: DT 99.96%, RF 99.84%, AdaBoost 99.77%; high ROC-AUC (99.96%)	Robustness evaluation on emerging threats and online streaming data needed
Filiz et al. (2024)	VAR, VECM, LSTM, GRU	Network traffic (digital media provider, Turkey)	LSTM & GRU best with $R^2 = 0.98$ and MAPE = 8.95%; strong temporal pattern recognition	Extend to multi-source network data and real-time predictive analytics
Sridevi et al. (2023)	Hybrid DNN + feature-engineered patterns	User activity logs from multiple companies	Achieved 96.3% ACC; effective in detecting insider threats; outperformed existing methods	Future work: scaling to larger enterprise networks and adaptive learning for evolving insider threats
Bokolo, Jinad & Liu (2023)	RF, DT, SVM, KNN, SGD, LR, NB, DL	Malware bytecodes, section codes, opcodes	DL achieved 96% ACC; outperformed traditional ML models	Explore real-time malware detection and robustness against obfuscation

III. METHODOLOGY

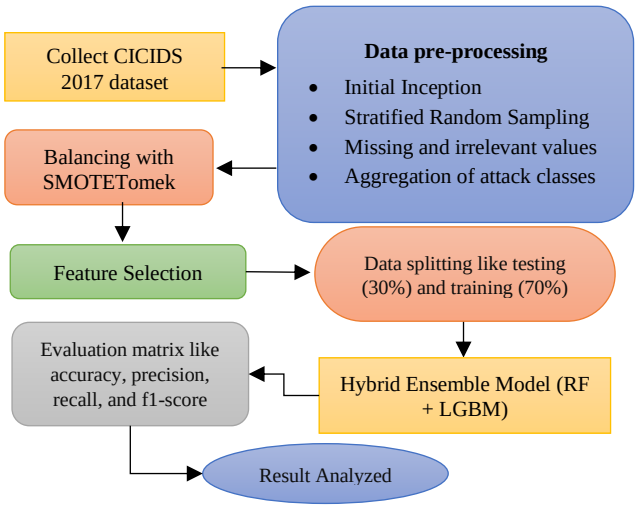

Fig. 1. Flowchart for Intrusion Detection Using the CICIDS 2017 Dataset

In the methodology for predictive analytics on intrusion detection using the CICIDS 2017 dataset, beginning with preprocessing that includes stratified random sampling, handling missing and irrelevant values, and aggregating attack classes. Categorical features are encoded using label encoding, and SMOTETomek is applied to balance class distribution. The features that are most important are chosen for retention through feature selection using Pearson correlation and mutual information. The dataset is divided into two parts: one for testing and one for training. For optimal intrusion detection performance, a hybrid ensemble model is trained and assessed using F1, REC, ACC, and PRE. This model combines RF and LGBM. The entire methodological framework is illustrated in Figure 1.

Briefly discussed below are the following steps of the proposed methodology:

A. *Data Collection*

The Canadian Institute for Cybersecurity's realistic testbed environment was the sole source for the extensive collection of network traffic activities that make up the CIC-IDS2017 dataset. Starting on July 3, 2017, and ending on July 7, 2017, the data encompasses several assault situations in addition to benign traffic. It covers different attack categories, including DoS, DDoS, Ports can, Web Attacks, Infiltration, Botnets, and Brute Force, providing valuable insights into intrusion detection and anomaly detection research. Some of the open problems identified in the data set based on the EDA are also discussed:

Figure 2 shows a correlation heatmap that shows the relationships among the invasion detection model's usage of several network traffic features. The attribute-to-feature correlation coefficient, with possible values between -1 and +1, is displayed in each heatmap cell. The deeper tones are

used to illustrate variegated characteristics when positive correlations are high, whereas light or negative numbers represent inverseness.

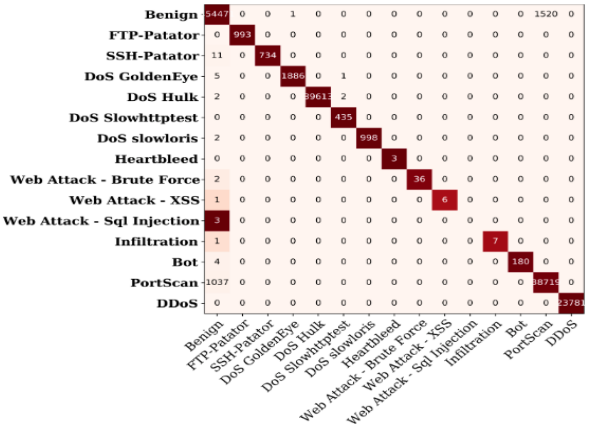

Fig. 2. Correlation Heatmap

B. Data Preprocessing

Pre-processing of the CIC-IDS2017 data is essential because of the existence of redundant features, imbalance in classes and irrelevant information. The steps that were followed in order to achieve the quality of data and its appropriateness in the classification work were as follows:

- **Initial Inspection:** The dataset was first analyzed with such functions as head, info, description to have a better idea of its structure, data types, and summary statistics. The inconsistencies or anomalies were also realized in the course of this examination [23].
- **Stratified Random Sampling:** As the dataset has large CSV files a stratified random sample was used to extract a small portion of the records so that representative subsets of the experiments can be made and minimal computational burden is minimized.
- **Handling Missing and Irrelevant Values:** Data integrity was ensured by removing missing values (NaN) and infinite entries, duplicate records, and constant features. Moreover, columns that are not relevant like Dst IP, Src IP, Src Port, Flow ID, and Timestamp were removed because they do not add to the process of detecting the attacks [24].
- **Aggregation of Attack Classes:** To simplify the classification problem, all attack types were aggregated into a single anomalous class, while normal traffic was labeled as 0. This transformation produced a binary classification problem, facilitating more robust APT detection.

C. Balancing Classes with SMOTETomek

SMOTETomek is a data pre-processing method used to solve class imbalance in classification. Figure. 3) shows that SMOTETomek was used on the CIC-IDS2017 data to achieve a balanced sample size between normal (0) and anomalous (1) classes [25]. The algorithm combines SMOTE that relies on similarities in feature space to artificially add representatives of the minority group with Tomek connections that drop samples that are borderline samples near decision boundaries. The hybrid strategy enhance the performance of the classifiers since the distribution of classes more balanced and there a lesser tendency in favour of the majority class.

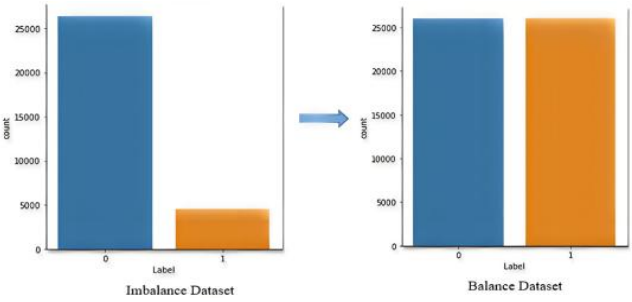

Fig. 3. Balancing with SMOTETomek

D. Feature Selection

Highly correlated features (correlation > 0.9) were removed using Pearson's correlation, reducing redundancy and improving training efficiency. The target variable's most pertinent features were subsequently selected using mutual information, resulting in the retention of the top 30 features out of 36 for classification [26].

E. Data splitting

A training subset and a testing subset comprise the CICIDS-2017 dataset. The training set makes up around 70% of the dataset, whereas the testing set makes up around 30%.

F. Classification with Hybrid Ensemble (RF+LGBM) Model

The RF algorithm is a robust ensemble learning method that uses bootstrap sampling to create many DT, which are then combined to produce final predictions. It reduces overfitting and increases tree diversity by randomly distributing features at each split. Equation (1) uses majority voting to provide predictions for classification:

$$H(x) = mode\{h_1(x), h_2(x), \ldots . h_T(x)\} \quad (1)$$

The t-th tree's forecast is denoted as $h_T(x)$, where T is the total number of trees.

LightGBM is a scalable and fast gradient boosting system. Its use of a leaf-wise growth strategy, as opposed to the level-wise growth in classic GBDT, results in a more effective reduction of loss. Its histogram-based splitting accelerates training by discretizing continuous values. The iterative boosting model is expressed as in Equation (2):

$$\hat{y}^{(t)}(x) = \hat{y}^{(t-1)}(x) + \eta h_t(x) \quad (2)$$

in which the learning rate is represented by η, and the iteration t tree is $h_t(x)$. The objective is made up of a regularization term, Ω, and the loss function, L.

$$\mathcal{L} = \sum_{i=1}^{n} L(y_i, \hat{y}^{(t)}(x_i)) + \sum_{t=1}^{T} \Omega(h_t) \quad (3)$$

This design enables LightGBM to cope with large volumes of high dimensional data with maintaining a high degree of predicted ACC. It is derived in Equation (3).

The hybrid ensemble makes use of the respective advantages of RF and LGBM. RF serves as a stabilizing and resistant factor in terms of overfitting, whereas LGBM is a more efficient and predictive factor in terms of ACC. The two models are trained separately and their results are combined using majority voting [27]. It is explaining in Equation (4):

$$H_{Hybrid}(x) = mode\{H_{RF}(x), H_{LGBM}(x)\} \quad (4)$$

The hybrid model is also suitable when dealing with difficult classification problems because this design is better at generalizing, and it is more resilient to noisy or imbalanced data.

G. Evaluation Metrics

The effectiveness of the classification models applied to the CICIDS 2017 dataset is determined by a confusion matrix that captures the results of both the projected and actual classifications [28]. The matrix consists of four important components. The confusion matrix is listed in below:

- **True Positive (TP):** A malicious attack has been positively recognized in the sample.
- **False Positive (FP):** A common sample was erroneously thought to be an attack.
- **False Negative (FN):** The system mistakenly recognized an attack sample as authentic communication.
- **True Negative (TN):** A normal sample was correctly identified as regular traffic. The performance metrics are as follows:

1) Accuracy

The percentage of correctly identified classes across all samples is directly correlated with the value. This metric is commonly employed to assess an IDS's efficiency in a balanced dataset, as shown in Equation (5).

$$Accuracy = \frac{\text{TP+TN}}{\text{TP+Fp+TN+}FN} \quad (5)$$

2) Precision

The value is directly proportional to the accuracy rate of the attack sample predictions as a percentage of the total attack sample counts. The ACC is calculable as showed by Equation (6).

$$Precision = \frac{\text{TP}}{\text{TP+FP}} \quad (6)$$

3) Recall

a ratio of the number of attack samples that were correctly predicted to the total number of attack samples. This measure is also known as the Detection Rate. With the aid of Equation (7), the REC can be obtained.

$$Recall = \frac{\text{TP}}{TP+FN} \quad (7)$$

4) F1-score

The PRE and REC are harmonically averaged, rendering them equal. Use it to show how much the two measures are out of whack to figure out if the solution is balanced, and have a better system critique. This metric is known by several names in Equation (8), including the F-Score and the F1.

$$F1-Score = \frac{2(Precision*Recall)}{\text{Precision+Recall}} \quad (8)$$

5) ROC

The ROC curve is a helpful graphical tool for assessing a classifier's performance because it shows the trade-off between the TPR and FPR at various threshold points. The curve is plotted on the x-axis as FPR and the y-axis as TPR with the change in sensitivity and specificity of the classifier as the threshold varies. The total capability of the model to differentiate between classes is the area of this curve (AUC).

The machine learning models are determined using these matrices.

IV. Result Analysis and Discussion

This section presents the experimental results of ML models which are used in Network Intrusion detection in CICIDS 2017 dataset and have performed all the tests on the computer with the Intel(R) Xeon(R) Gold 6248R processor, NVIDIA GeForce RTX 3090 graphics card and Windows 10 operating system. The following measures are used to access the performance: F1, REC, ACC, and PRE. The Hybrid Ensemble model results in Network Intrusion detection are availed in Table II The Hybrid Ensemble (RF + LGBM) model proposed was excellent as its ACC and PRE were 99.90 and 99.80, respectively, and REC and F1 were 99.92 and 99.90, respectively.

TABLE II. Performance of Hybrid Model on The CICIDS 2017 Dataset for Network Intrusion Detection

Performance Measures	Hybrid Ensemble (RF+ LGBM) Model
Accuracy	99.90
Precision	99.80
Recall	99.92
F1-score	99.90

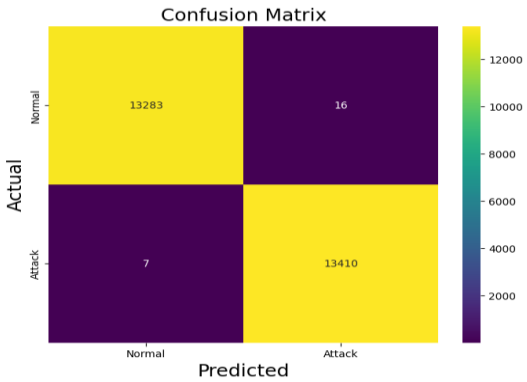

Fig. 4. Confusion Matrix of Hybrid Ensemble Model

The CICIDS2017 dataset's confusion matrix, which illustrates how successfully a hybrid ensemble model identified observations as "Normal" or "Attack," is displayed in Figure 4. For the 'Normal' class, the model correctly classified 13,283 instances, with only 16 misclassified as 'Attack'. For the 'Attack' class, 13,410 instances were correctly identified, while just 7 were incorrectly labeled as 'Normal'.

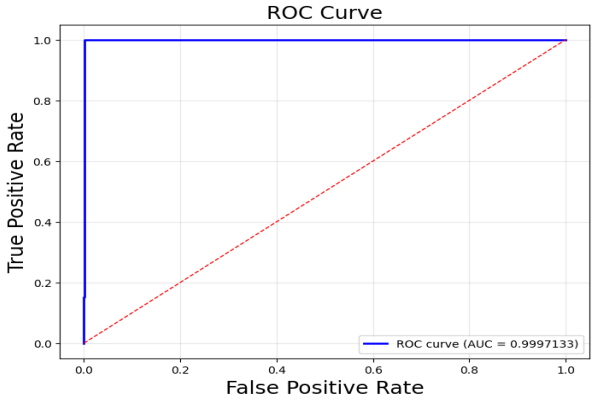

Fig. 5. ROC Graph of Hybrid Ensemble Model

Figure 5 displays the ROC curve, which illustrates the effectiveness of the hybrid ensemble model for binary classification of the CICIDS 2017 dataset, particularly for the 'Normal' and 'Attack' classes of the supply chain network. At

various thresholds for categorization, the ROC curve shows how the TPR and FPR relate to one another. The AUC is exceptionally high at 0.9997, quantifying the model's outstanding discriminatory power and indicating that the hybrid ensemble model performs with near-perfect ACC in separating the two classes represented in the dataset.

A. Comparative Analysis and Discussion

Presented here is the comparative analysis of Network Intrusion detection using the CICIDS 2017 dataset. Table III shows the results of a comparative evaluation of the three models that were used for network IDS on the CICIDS 2017 dataset: Hybrid Ensemble (RF+LGBM), NB, and SVM. ACC, PRE, REC, and F1 were all 99.90% for the Hybrid Ensemble model, making it the most performing model overall. NB follows with a high sensitivity but low PRE indicator (ACC: 72.96%, PRE: 65.76%, REC: 96.71%, and F1: 78.29%). With an F1 of 89.94%, ACC of 89.67%, PRE of 87.63%, and REC of 92.49%, the SVM model performs similarly to the Hybrid Ensemble but with less effectiveness.

TABLE III. ML MODELS COMPARISON ON THE CICIDS 2017 DATASET FOR NETWORK INTRUSION DETECTION

Performance Measures	**Hybrid Ensemble (RF+ LGBM) Model**	**NB** [29]	**SVM** [30]
Accuracy	99.90	72.96	89.67
Precision	99.80	65.76	87.63
Recall	99.92	96.71	92.49
F1-score	99.90	78.29	89.94

The suggested findings indicate that ensemble-based models, in particular, the Hybrid Ensemble (RF+LGBM), achieve higher results in network intrusion detection with the assistance of the CICIDS 2017 dataset. The Hybrid Ensemble model had 99.90% ACC, which was best among other models. Its high level of reliability and robustness was verified by its high PRE, REC, and F1. These findings validate the applicability of ensemble learning techniques in particular the Hybrid Ensemble to deal with high dimensional and complex tasks within an intrusion detection system.

V. CONCLUSION AND FUTURE WORK

The contemporary digital terrain is increasingly posing more sophisticated and serious cyber threats which jeopardize the traditional network defines systems. The machine learning-based intrusion detection systems have become powerful tools to handle such threats since they are capable of identifying abnormalities and attack patterns during real-time. The paper demonstrates that the ensemble learning techniques may be utilized successfully to enhance the ACC and power of intrusion detection systems. The proposed Hybrid Ensemble Model (RF + LGBM) showed excellent performance on CICIDS 2017, obtaining an ACC of 99.90, a PRE of 99.80, a REC of 99.92, and an F1 of 99.90. Such results show the ability of the model to generalize the various forms of attacks with less false positive and false negative. SMOTETomek balancing, selection of features (through Pearson correlation and mutual information) and hybrid ensemble learning also helped with better detection and lower computation cost.

Future research will apply the suggested framework to multiclass recognition, maintaining the variety of attack types instead of simplifying it to a binary task. Real-time and streaming detection will also be discussed to evaluate the performance with the constant network traffic to address the issues of deployment in enterprise and IoT settings. Moreover, explainable AI techniques like SHAP values will enhance the interpretability, which allows security analysts to learn more about model decisions. "

REFERENCES

[1] A. I. Jony and S. A. Hamim, "Navigating the Cyber Threat Landscape: A Comprehensive Analysis of Attacks and Security in the Digital Age," *J. Inf. Technol. Cyber Secur.*, vol. 1, no. 2, pp. 53–67, Dec. 2023, doi: 10.30996/jitcs.9715.

[2] S. Narang and V. G. Kolla, "Next-Generation Cloud Security: A Review of the Constraints and Strategies in Serverless Computing," *Int. J. Res. Anal. Rev.*, vol. 12, no. 3, pp. 1–7, 2025, doi: 10.56975/ijrar.v12i3.319048.

[3] R. Q. Majumder, "Machine Learning for Predictive Analytics: Trends and Future Directions," *Int. J. Innov. Sci. Res. Technol.*, vol. 10, no. 4, pp. 3557–3564, May 2025, doi: 10.38124/ijisrt/25apr1899.

[4] G. Apruzzese *et al.*, "The Role of Machine Learning in Cybersecurity," *Digit. Threat. Res. Pract.*, vol. 4, no. 1, pp. 1–38, Mar. 2023, doi: 10.1145/3545574.

[5] N. K. Prajapati, "Federated Learning for Privacy-Preserving Cybersecurity: A Review on Secure Threat Detection," *Int. J. Adv. Res. Sci. Commun. Technol.*, vol. 5, no. 4, pp. 520–528, Apr. 2025, doi: 10.48175/IJARSCT-25168.

[6] S. Thangavel, S. Srinivasan, S. B. V. Naga, and K. Narukulla, "Distributed Machine Learning for Big Data Analytics: Challenges, Architectures, and Optimizations," *Int. J. Artif. Intell. Data Sci. Mach. Learn.*, vol. 4, no. 3, pp. 18–30, Oct. 2023, doi: 10.63282/3050-9262.IJAIDSML-V4I3P103.

[7] A. R. Bilipelli, "AI-Driven Intrusion Detection Systems for Large-Scale Cybersecurity Networks Data Analysis : A Comparative Study," *TIJER – Int. Res. J.*, vol. 11, no. 12, pp. 922–928, 2024.

[8] H. Kali, "The Future of HR Cybersecurity: AI-Enabled Anomaly Detection in Workday Security," *Int. J. Recent Technol. Sci. Manag.*, vol. 8, no. 6, pp. 80–88, 2023.

[9] F. Alserhani and A. Aljared, "Evaluating Ensemble Learning Mechanisms for Predicting Advanced Cyber Attacks," *Appl. Sci.*, vol. 13, no. 24, p. 13310, Dec. 2023, doi: 10.3390/app132413310.

[10] S. Narang and A. Gogineni, "Zero-Trust Security in Intrusion Detection Networks: An AI-Powered Threat Detection in Cloud Environment," *Int. J. Sci. Res. Mod. Technol.*, vol. 4, no. 5, pp. 60–70, Jun. 2025, doi: 10.38124/ijsrmt.v4i5.542.

[11] W. S. Admass, Y. Y. Munaye, and A. A. Diro, "Cyber security: State of the art, challenges and future directions," *Cyber Secur. Appl.*, vol. 2, 2024, doi: 10.1016/j.csa.2023.100031.

[12] D. Patel, "Zero Trust and DevSecOps in Cloud-Native Environments with Security Frameworks and Best Practices," *Int. J. Adv. Res. Sci. Commun. Technol.*, vol. 3, no. 3, 2023.

[13] V. M. L. G. Nerella, K. K. Sharma, S. Mahavratayajula, and H. Janardhan, "A Machine Learning Framework for Cyber Risk Assessment in Cloud-Hosted Critical Data Infrastructure," *J. Inf. Syst. Eng. Manag.*, vol. 10, no. 4, pp. 2409–2421, 2025.

[14] R. Panigrahi and S. Borah, "A detailed analysis of CICIDS2017 dataset for designing Intrusion Detection Systems," *Int. J. Eng. Technol.*, 2018.

[15] P. G. A. N. Gowda, "Advancing Predictive Analytics in Cybersecurity: Utilizing GRU-Capsule Networks for Financial Threat Detection," in *2025 International Conference on Machine Learning and Autonomous Systems (ICMLAS)*, IEEE, Mar. 2025, pp. 1074–1081. doi: 10.1109/ICMLAS64557.2025.10967823.

[16] R. Fu, "Design and Implementation of Network Intrusion Detection System based on Machine Learning," in *2025 International Conference on Intelligent Systems and Computational Networks (ICISCN)*, 2025, pp. 1–6. doi: 10.1109/ICISCN64258.2025.10934502.

[17] R. O. Ogundokun, P. A. Owolawi, and E. Van Wyk, "LiteRT-IDSNet: A Lightweight Hybrid Deep Learning Framework for Real-Time Intrusion Detection in Industrial IoT Using the RT-IoT 2022 Dataset," in *2025 60th International Scientific Conference on Information, Communication and Energy Systems and Technologies (ICEST)*, IEEE, Jun. 2025, pp. 1–4. doi:

10.1109/ICEST66328.2025.11098207.

[18] F. Zhang, "Network Intrusion Detection System Based on Separable Convolution AutoEncoder with Long Short-Term Memory," in *2024 International Conference on Integrated Intelligence and Communication Systems (ICIICS)*, 2024, pp. 1–5. doi: 10.1109/ICIICS63763.2024.10859437.

[19] F. A. Khan *et al.*, "Balanced Multi-Class Network Intrusion Detection Using Machine Learning," *IEEE Access*, vol. 12, pp. 178222–178236, 2024, doi: 10.1109/ACCESS.2024.3503497.

[20] G. Filiz, A. Yıldiz, E. Kara, S. Altıntaş, and T. Çakar, "Artificial Intelligence Driven Multivariate Time Series Analysis of Network Traffic Prediction," in *2024 Innovations in Intelligent Systems and Applications Conference (ASYU)*, IEEE, Oct. 2024, pp. 1–4. doi: 10.1109/ASYU62119.2024.10756993.

[21] D. Sridevi, L. Kannagi, V. G, and S. Revathi, "Detecting Insider Threats in Cybersecurity Using Machine Learning and Deep Learning Techniques," in *2023 International Conference on Communication, Security and Artificial Intelligence (ICCSAI)*, 2023, pp. 871–875. doi: 10.1109/ICCSAI59793.2023.10421133.

[22] B. Bokolo, R. Jinad, and Q. Liu, "A Comparison Study to Detect Malware using Deep Learning and Machine learning Techniques," in *2023 6th International Conference on Big Data and Artificial Intelligence, BDAI 2023*, 2023. doi: 10.1109/BDAI59165.2023.10256957.

[23] N. Prajapati, "The Role of Machine Learning in Big Data Analytics: Tools, Techniques, and Applications," *ESP J. Eng. Technol. Adv.*, vol. 5, no. 2, pp. 16–22, 2025, doi: 10.56472/25832646/JETA-V5I2P103.

[24] R. Patel, "Automated Threat Detection and Risk Mitigation for ICS (Industrial Control Systems) Employing Deep Learning in Cybersecurity Defence," *Int. J. Curr. Eng. Technol.*, vol. 13, no. 06, pp. 584–591, Dec. 2023, doi: 10.14741/ijcet/v.13.6.11.

[25] V. Shah, "Analyzing Traffic Behavior in IoT-Cloud Systems : A Review of Analytical Frameworks," *Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol.*, vol. 9, no. 3, pp. 877–885, 2023.

[26] V. Shah, "Managing Security and Privacy in Cloud Frameworks: A Risk with Compliance Perspective for Enterprises," *Int. J. Curr. Eng. Technol.*, vol. 12, no. 06, pp. 1–13, 2022, doi: 10.14741/ijcet/v.12.6.16.

[27] V. Thangaraju, "Enhancing Web Application Performance and Security Using AI-Driven Anomaly Detection and Optimization Techniques," *Int. Res. J. Innov. Eng. Technol.*, vol. 9, no. 3, 2025, doi: 10.47001/IRJIET/2025.903027.

[28] P. Vanin *et al.*, "A Study of Network Intrusion Detection Systems Using Artificial Intelligence/Machine Learning," *Appl. Sci.*, vol. 12, no. 22, p. 11752, Nov. 2022, doi: 10.3390/app122211752.

[29] T.-H. Chua and I. Salam, "Evaluation of Machine Learning Algorithms in Network-Based Intrusion Detection Using Progressive Dataset," *Symmetry (Basel).*, vol. 15, no. 6, p. 1251, Jun. 2023, doi: 10.3390/sym15061251.

[30] D. Han, H. Li, X. Fu, and S. Zhou, "Traffic Feature Selection and Distributed Denial of Service Attack Detection in Software-Defined Networks Based on Machine Learning," *Sensors*, vol. 24, no. 13, 2024, doi: 10.3390/s24134344.

