Adaptive Ensemble Models for Robust Intrusion Detection in Cloud Environment on Imbalanced Dataset

Swarnalatha K[1], Nirmalajyothi Narisetty[2]*, Gangadhara Rao Kancherla[3],
Neelima Guntupalli[4], Simhadri Mallikarjuna Rao[5], Archana Kalidindi[6]
Dept. of CSE, Acharya Nagarjuna University, Guntur, 522510, India[1, 3, 4]
Dept. of CSE, Koneru Lakshmaiah Education Foundation, Bowrampet, Hyderabad, Telangana, 500043, India[2]
Dept. of IT, Vasireddy Venkatadri International Technology, Nambur, India[5]
Dept. of CSE-AIML & IoT, VNR Vignana Jyothi Institute of Engineering and Technology, Hyderabad, India[6]

***Abstract*—The rapid development of Information storage and sharing technologies brings new challenges in protecting against network security attacks. In this study, ensemble learning models are evaluated to enhance the performance of a network intrusion detection system (NIDS) with three phases through machine learning approaches. In the first phase, the unbalanced dataset is processed through four re-sampling techniques, such as SMOTE, RUS, RUS+ROS, and RUS+SMOTE, for balancing treatment. In the second phase, Random Forest feature selection is imposed for these four balanced datasets. Finally, three Ensemble Models named as EM1, EM2 and EM3 are designed using six basic classifiers and thus evaluated. In earlier studies, the first and second phases were evaluated through an SVM binary classifier for four feature subsets. The four feature subsets are obtained through Random Forest feature selection with the four different thresholds of Cumulative Feature Importance Scores (CFIS) (85%, 90%, 95% and 99%). With the observation of the evaluated results, three challenges were identified: i) The highest accuracy obtained through the re-sampling method required maximum computational time. ii) Different thresholds of CFIS exhibit instability in performance metrics as well as computational times, even though the number of features is less. iii) The adopted multi-class SVM classifier's efficiency to detect the attacks within minimum computational time and without compromising accuracy when compared to earlier works is yet to be ascertained. In this study, an attempt has been made to address these challenges with ensemble learning. Three ensemble models are chosen for the evaluation process conducted on the adopted CIIDS -2017 dataset. Finally, the comparative results are presented, and decisive discussions are carried out for implementing the prevention and mitigation algorithms by security professionals.**

Keywords*—Resampling methods; cloud computing; feature selection; ensemble model; intrusion detection system; machine learning

I. INTRODUCTION

The rapid evolution of the cloud computing technology facilitates the infrastructure, network and applications through Internet in the present era. The widespread of cloud computing usage has reduced the management and maintenance costs of IT infrastructure. Cloud resources are managed by different organizations in the network using standards and methods [1]. This has driven its efficient business model and scalability, and at the same time has introduced challenges in network security. Due to its decentralized structure and centralized management, IT infrastructure is vulnerable to contemporary attacks [2].

The biggest obstacle to the success and use of cloud computing by an organization or individual is its security. The most common serious threats in cloud attacks are Distributed Denial of Service (DDoS), which causes excessive volume of traffic from multiple distributed systems, leading to denial of service to users. In DDoS attacks, unauthorized actions aim to temporarily or permanently disrupt the services of network members against counterattacks such as cloud services, thus encouraging the use of cloud computing [3]. The impacts of a successful DDoS attack are varied and can be severely damaging. Organizations may incur significant financial losses from service interruptions, as well as expenses related to incident response and system restoration efforts. Finally, DDoS attacks can cause customers to lose trust in certain products, affecting performance and brand reputation [4] [5].

To maintain the availability and reliability of network services, the research community and industry sectors dedicate significant efforts to developing Intrusion Detection Systems (IDS). Ongoing research by the Cloud Security Alliance (CSA) has identified DDoS attacks as some of the most prevalent threats to cloud security [6]. For example, prominent studies referenced in [7] [8] have concentrated on developing advanced strategies to counter these attacks effectively. Despite the widespread adoption of IDS, DDoS attacks are often avoided in many scenarios. With the growing volume of data constantly transmitted between networks, IDS has proven effective in detecting disruptions within large datasets [9].

Many researchers are working on this to provide a solution in identifying these types of attacks in network intrusion detection based on the available static datasets. Most of these datasets are skewed. Class imbalance in cloud security attacks can influence the performance of intrusion detection systems (IDSs). Since legitimate traffic samples are far beyond malicious, machine learning models are often affected by these groups, resulting in lower detection rates for several types of attacks. This flaw limits the ability of IDSs to detect

rare but important threats, thwarts attack plans, and compromises the security standards of the entire system. Not only in Intrusion detection, other areas like Natural Language Processing [10], [11], Image recognition [12], genetic engineering [13], [10], financial fraud detection [14], web mining to text categorization [15] also have been advocated imbalanced data classification. Therefore, addressing class imbalance is crucial to improve detection accuracy and strengthen defense mechanisms in cloud environments. The quality of the dataset is very significant in the classification method, and certain imbalance classes dominate the dataset.

Ensemble methods proven to be effective in increasing the efficacy of intrusion detection systems (IDS). These methods combine the strength of several machine learning models to achieve improved predictive performance and robustness compared to individual models. Combining various techniques improves the detection by aggregating multiple instances. It also reduces the Variance and biases associated with the limitations of a single integration, especially in high dimensions and skewed data [16] [17].

After going through the above research works, the three-phase ensemble models are designed and evaluated. The coordination methods, such as weighted voting, can mitigate this problem by evaluating the impact of a limited class, thereby reducing the negativity and improving the detection of less resilient types. In addition, to combine re-sampling methods, feature selection with traditional machine learning techniques may be generalize and effective.

The results in [18] demonstrate the performance of re-sampling methods for NIDS. RUS+SMOTE achieves the smallest computational time but delivers the lowest performance. SMOTE achieves the highest accuracy, precision, recall, and F-measure across all CFIS thresholds. In spite of its high accuracy of 99.26% and F-Measure of 91, the (RUS+ROS) is less efficient due to its time complexity of 4808 sec.

To enhance the performance of [18], the current study, which uses ensemble learning, aims to develop a stable and generalized model with low time complexity for real-time imbalanced datasets. The study has been carried out with the following key contributions:

- Study the different re-sampling methods for imbalance treatment to enhance the classification results.
- To develop and evaluate the ensemble classification models from base classifiers for improving the performance of minority class samples.
- To analyze the generalized three-phase ensemble models' behavior in mitigating the class imbalance and improving the AUC.
- Different ensemble models built by the same basic classifier set for comparative analysis.

The rest of the study is organized as follows: Section II provides a summary of the various research efforts in this field. A description of the methodology of the proposed ensemble models is presented in Section III. Section IV explains the analysis of the experimental results. Finally, Section V concludes the current study and discusses future perspectives.

II. Related Work

There have been various studies that have led to the development of automatic intrusion detection in network communication. Various Machine Learning and Deep Learning methods were used to generate the hybrid and ensemble models for dealing with unbalanced data to detect these types of attacks. Some of them are outlined below.

A survey conducted in [19] examined the various strategies for addressing imbalanced data when detecting network intrusions. The common strategies that can be applied to balance the instances and various approaches that can be used to mitigate the challenges of an imbalanced dataset, such as data level methods, algorithm level methods and fusion methods, were explored. The commonly used oversampling techniques in the literature, like SMOTE and ADASYN, were considered. The findings indicate that ADASYN generally outperforms SMOTE in terms of F1-score and recall, while SMOTE may be suitable for maintaining high accuracy.

The authors in [20] aimed to address the challenge of high false negative rates and improve the predictability of minority classes in NIDS for both binary and multiclass classification. There are three stages in the proposed strategy: fine-tuning the training and testing subset distributions, selecting features, and using class weights. The experiments were conducted using the well-established NSL KDD and UNSW-NB15 datasets for binary classification as well as multi-class classification. Multiple models were generated using an effective refinement strategy aimed at reducing False Negative Rates (FNR) and increasing minority class predictability. Based on the results, it was demonstrated that it is possible to achieve a satisfactory trade-off between FNR, accuracy, and minority detection with the proper parameters. While the model's performance on a limited dataset raises concerns related to generalization. This can be addressed in future by exploring the efficacy of the model on large datasets in enhancing the performance on minority attack detection.

Hongpo Zhang et al. proposed a two-stage intrusion detection model for IoT security [21]. Stage 1 employs LightGBM for efficient initial classification, while Stage 2 uses CNN for fine-grained anomaly detection. The model effectively addresses class imbalance and achieves high accuracy, F1-score, and MCC. It outperforms state-of-the-art methods in terms of efficiency and making it a promising solution for large-scale data. Future work will focus on improving feature selection and imbalance handling techniques.

The authors in [22] aimed to address the challenge of high false negative rates and improve the predictability of minority classes in NIDS. In their approach, the training and testing subset distributions are adjusted, features are selected, and class weights are used. Both binary and multi-class classification experiments were conducted with the NSL KDD and UNSW-NB15 datasets. Several models were developed using an effective refinement strategy to reduce False

Negative Rates (FNR) and increase minority class prediction accuracy. It was demonstrated by the results that proper parameters can be utilized to achieve an acceptable trade-off between FNR, accuracy, and minority detection.

The study conducted in [23] investigates the impact of class imbalance countermeasures on model interpretability for both AI users and experts. Despite their effectiveness in improving prediction performance, many countermeasures often compromise interpretability. Interpretability is only preserved by feature selection and cost-sensitive approaches, according to our experiments. Normally, re-sampling and most classification algorithms cannot be used in settings where gaining knowledge and being able to interpret it are essential. Several guidelines are provided for selecting interpretable countermeasures, and we highlight future research opportunities.

The BMCD algorithm has been proposed for large-scale multiclass intrusion detection datasets in [24]. With BMCD, minority class detection is improved by adapting SMOTE to multiclass situations. On a combined dataset from CICIDS2017, BMCD is found to improve intrusion detection performance over existing methods in addressing class imbalance and addressing class imbalance. With BMCD-balanced datasets, random forest classifiers, in particular, perform significantly better.

Pieter Barnard et al. proposed a two-staged pipeline framework in [25] for robust network intrusion detection. The first stage leverages a powerful machine learning model, namely XGBoost. To understand the model's decision-making process, we employ the SHAP framework to generate explanations of its predictions. The second stage utilizes these explanations to train an autoencoder. This autoencoder acts as an anomaly detector, specifically designed to identify unseen attacks not encountered during the initial training. The evaluation is carried out on the NSL-KDD dataset and demonstrates its effectiveness in accurately detecting new attacks. It is also compared to various state-of-the-art intrusion detection methods.

The effect of the class imbalance problem may lead to biased model performance. To address this, the authors Ngan Tran et al. have explored various techniques for handling the class imbalance in NIDS [26]. Among them, the downsampling + upsampling + SMOTE (DUS) was the most effective re-sampling technique based on the experimental results conducted on the NSL-KDD dataset. An Ensemble model combined with DUS outperformed other machine learning classifiers. The impact of the number of classes on model performance is also evaluated, and discovered that more imbalanced classes negatively affect accuracy. Finally, researchers highlighted the importance of addressing class imbalance in NIDS and provided insights into the effective techniques. By understanding the impact of class imbalance and employing appropriate strategies, researchers can improve the performance of NIDS models in real-world scenarios.

In [27], the authors introduced a novel DNN-based approach for detecting DoS/DDoS attacks, which utilizes three imbalanced datasets of NIDS: CICIDS2017, CSE-CICIDS2018, and CICDDoS2019. To mitigate class imbalance, a K-means based technique is employed to generate semi-balanced datasets. Feature selection is performed using Linear Discriminant Analysis, and four metaheuristic algorithms (AIS, FA, IWO, and CS) are integrated with DNN to enhance performance. Experimental results demonstrate the effectiveness of the proposed approach, particularly AIS-DNN, which achieves high accuracy (up to 99.99%) and outperforms existing methods.

According to [28], the quality of the dataset is very important to improve minority class attack detection in NIDS. In this study, a combined oversampling and undersampling technique is used, followed by training with deep learning models. The proposed approach significantly enhances the detection rate of minority classes, with the CNN model achieving 99.8% accuracy in binary classification and the MLP model achieving 99.9% accuracy in multi-class classification. These results offer a promising direction for improving NIDS and detecting minority class attacks.

The authors in [29] demonstrated a novel autoencoder-based anomaly detection method for IoT security. By effectively utilizing packet metadata and training on normal data, the method accurately detects anomalies in encrypted IoT traffic. It outperforms traditional methods, is versatile, and can handle various encryption protocols, making it a robust solution for IoT security. However, the authors have evaluated the performance on only one type of attack.

The Data Generative Model to detect anomalies in an imbalanced dataset was proposed in [30]. Also concentrated on certain types of attacks where those were underrepresented. To address this, the present study proposed a Data Generative Model (DGM) using Conditional Generative Adversarial Networks (CGAN) to increase the number of minority class samples in the CICIDS2017 dataset. The researcher concluded that their experiments demonstrated that the DGM effectively detected new attacks and significantly improved the weighted F1-score (99.12%) compared to existing methods.

Traditional intrusion detection systems often struggle to effectively identify rare but critical attacks due to class imbalance in network traffic [31]. This research introduces a novel framework, S2CGAN-IDS, to address this challenge. The framework categorizes network traffic into three imbalance levels: ample, scarce, and rare. To augment the dataset, it combines the Synthetic Minority Over-sampling Technique (SMOTE) and a novel Synthetically Controlled Generative Adversarial Network (SCGAN). SMOTE oversamples scarce class data, while SCGAN generates synthetic samples for the rare class. The augmented dataset is then fed into a simple Deep Neural Network (DNN) classifier, which effectively distinguishes between normal and anomalous traffic, including rare attacks.

The authors in [32] explained the importance of safeguarding the network to avoid malicious access. Recent advancements in deep learning have significantly improved NIDS performance. However, the inherent class imbalance in network traffic, where normal traffic significantly outweighs attack traffic, remains a significant challenge. To address this issue, this research proposes a hybrid approach that combines oversampling and under sampling techniques. Specifically,

Synthetic Minority Over-sampling Technique (SMOTE) is employed to increase the representation of minority classes, while Tomek Links is used to remove noisy samples. Additionally, two powerful deep learning models, Long Short-Term Memory (LSTM) and Convolutional Neural Network (CNN), are utilized to enhance the model's ability to capture intricate patterns in network traffic.

The imbalanced data in NIDS can pose significant challenges, which may lead to biased models that struggle to accurately classify minority classes. Intrusion Detection Systems (IDS) datasets often exhibit this imbalance, with common attacks like DDoS dominating the data. To address this issue, techniques like SMOTE and ADASYN can be used to create synthetic data for minority classes [34]. However, not all features within a dataset are equally important. Feature selection methods like Recursive Feature Elimination (RFE) can help to identify the most relevant features, improving model performance and reducing computational costs. Finally concluded that the Decision Trees consistently outperformed Random Forest and KNN in terms of intrusion detection.

III. Proposed Approach

This section explains the various steps involved in the methodology implementation of the proposed three-phase ensemble model, which is presented in Fig. 1. They are: i) data preparation and imbalance treatment, ii) Random Forest Feature selection and iii) Base classifiers evaluation and evaluating ensemble models. These phases are explained in detail below. The base classifiers are explained briefly in the next subsection.

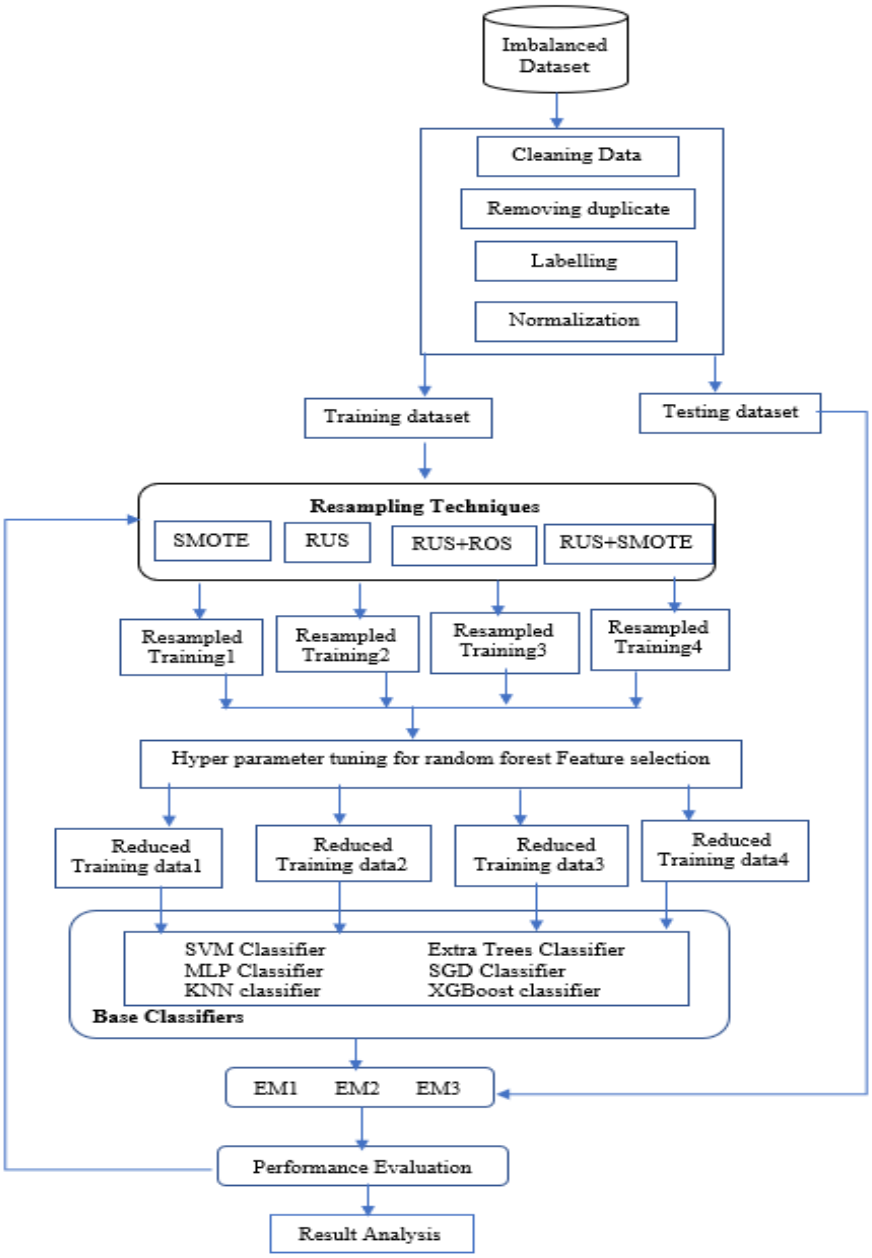

Fig. 1. Methodology of the three-phase ensemble model.

A. Base Classifiers

The construction of three ensemble models, EM1, EM2 and EM3, with the adoption of six base classifiers (BM).

1) Extra Tree classifier: The ET classifier excels in decision making by using the random splits threshold instead of the optimal split. In addition to speeding up the training process, this approach enhances the model's generalization capabilities, reducing overfitting. A key advantage of the Extra Trees Classifier is its ability to produce stable and accurate predictions by averaging outputs from multiple trees. These benefits, combined with its scalability and fast training time, make the Extra Trees Classifier a powerful and reliable tool in intrusion detection [35], and the graphical representation is shown in Fig. 2. It also still has the limitation that it overfits if the dataset contains too much noise.

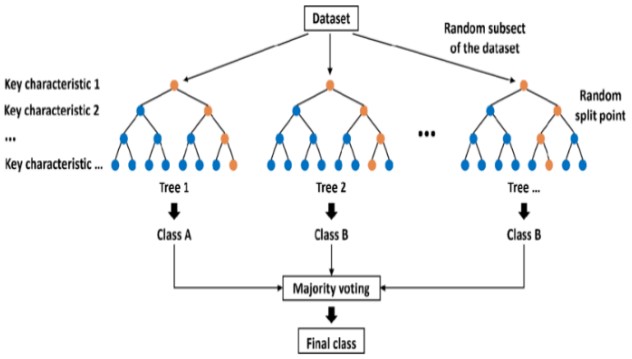

Fig. 2. Extra Tree Classifier.

By optimizing a loss function using gradient descent, it is known for its scalability and effectiveness [36].

2) Support Vector Machine (SVM): It is one of the popular classifiers in machine learning and is well known for its effectiveness in high-dimensional spaces as well as with non-linear decision boundaries. It excels in identifying the patterns among the data samples using various non-linear kernel functions by preventing the overfitting problem. Based on the assumption that all the above advantages will help to improve the performance of intrusion detection systems, the SVM has also been included in the investigation [37]. The following Fig. 3 shows the pictorial representation of the SVM classifier.

3) MLP classifier: It is a type of neural network and is capable of capturing complex, non-linear relationships in data. This makes it particularly valuable for detecting intricate patterns in the data, which are common in intrusion detection scenarios. The foundation of a neural network is a perceptron, which contains three layers of neurons, namely, the input layer, hidden layer and output layer. It uses backpropagation for improving the learning rate and decreasing the error rate [38]. The MLPs are flexible and can model intricate decision boundaries, providing additional power to the ensemble model.

4) KNN classifier: Because of its many benefits, the KNN classifier is often chosen for machine learning jobs. Its non-parametric nature enables it to handle datasets with irregular patterns efficiently, and its simplicity and ease of implementation make it perfect for learners. KNN is adaptable, works well for both regression and classification applications, and naturally manages multi-class issues. It doesn't need a training phase because it is a lazy learner, which makes it computationally efficient for creating models. It works well with tiny datasets and, by taking into account the local structure of the data, can manage non-linear decision limits. KNN also doesn't depend on presumptions about feature interactions and is resilient to outliers (if k is chosen correctly).

5) XG boost: With so many benefits, the XGBoost classifier is a popular and powerful machine learning technique. With features like tree pruning and parallel processing that condense computation times and improve scalability, it is incredibly effective and performance-optimized. XGBoost produces strong and trustworthy predictions even when dealing with big datasets and missing values. It is perfect for complex models since regularization techniques like L1 and L2 can reduce overfitting. It is appropriate for a variety of applications, such as recommendation systems, healthcare, and finance, due to its high performance, speed, and adaptability.

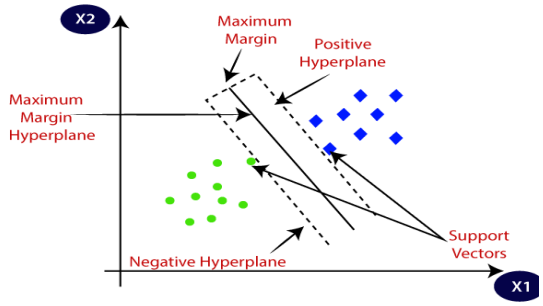

Fig. 3. Pictorial representation of an SVM classifier.

The main objective of selecting the above classifiers is to benefit from the diverse learning philosophy in building a robust generalized model towards improving the intrusion detection rate [33].

B. Phase 1

This subsection consists of two stages: Data preparation and Data imbalance treatment.

1) Data preparation: The preliminary stage of developing an Intrusion Detection System (IDS) includes gathering comprehensive data on network traffic that covers both benign and contemporary attacks. These types of attacks are resembled in the datasets CICIDS-2017. The complete data preparation steps, like data cleaning, eliminating duplicate rows and Label encoding, are carried out in line with [19]. A normalization step is performed to standardize feature magnitudes, enhancing the algorithm's robustness and ensuring no single feature dominates during learning.

2) Data imbalance treatment: Data imbalance in intrusion detection systems (IDS) is a critical issue that affects the system's ability to detect malicious activity effectively. Intrusion detection datasets often exhibit a significant imbalance, where normal (benign) activities vastly outnumber malicious (attack) events. Treating this imbalance is essential to improve the system's performance and reliability. The main advantages of balancing the dataset are enhancing model generalization, improving the detection rate and minority sample detection. To balance the class labels, the study utilizes four resampling methods, SMOTE, RUS, RUS+ROS and RUS+SMOTE, outlined by [34] and [18]. All four methods were applied to training data in parallel to generate respective balanced datasets.

C. Phase 2

In this phase, feature selection is carried out with the adoption of the Random Forest feature selection method as per the feature extraction process and hyperparameter tuning given in [18]. The purpose behind selecting different thresholds in the progressive approach is to conduct a systematic study of the balance between complexity, robustness and accuracy. The resulting datasets are fed as input to the feature selection phase to mitigate the overfitting problem and carried out based on CFIS score thresholds (85%, 90%, 95% and 99%) for extracting four feature subsets.

D. Phase 3

This classification phase consists of six base classifiers' evaluation, along with three ensemble models' evaluation.

1) Base classifiers evaluation: The results of the study [18] in terms of intrusion detection were promising, but there is still scope for optimizing the computational complexity as well as the performance to generate a stable model. The current study is conducted with the objective of addressing these aspects by building a generalized ensemble model. The four different datasets generated as output of feature selection phase are given as input to four base classifiers to generate each different ensemble model based on majority voting. The novelty of the present study generates an ensemble model with different datasets contrasting to the existing ensemble models, where all the base classifiers will be training with same dataset. Each time four classifiers are selected from the following base classifiers Extra Trees (ET), Stochastic Gradient Descent (SGD), Support Vector Machine (SVM), KNN, XGBoost and Multi-Layer Perceptron (MLP) to build the ensemble models.

2) Evaluation of EM1, EM2 and EM3: The EM1 model is constructed with Extra Tree, SGD, SVM, and XGBoost Classifiers. EM2 model created with KNN, SGD, SVM, and MLP Classifiers. The EM3 is built with Extra Tree, SGD, SVM, and MLP Classifiers. The strategic decision to select these six diverse base classifiers aimed at leveraging their complementary strength to generate more robust and accurate classifiers in classifying the network samples. The strengths or the advantages of these algorithms are discussed below. Voting is an ensemble learning technique that works better when the predictions are combined for a classification case. Much better performance is typically obtained by voting rather than using a single classifier [39].

Two kinds of voting are used in classification problems — hard voting [40] and soft voting [41]. Hard voting is to take the one with the most votes, while soft voting means averaging probabilities. The current study leveraged soft voting; its importance stems from its ability to improve prediction accuracy and robustness by leveraging the strengths of multiple models. The final prediction is based on the weighted average of these probabilities, leading to a more nuanced decision-making process. The output of the snippet for ensemble mode implementation is shown in Fig. 4. Finally, the pseudocode of the three-phase ensemble model is presented in Fig. 5.

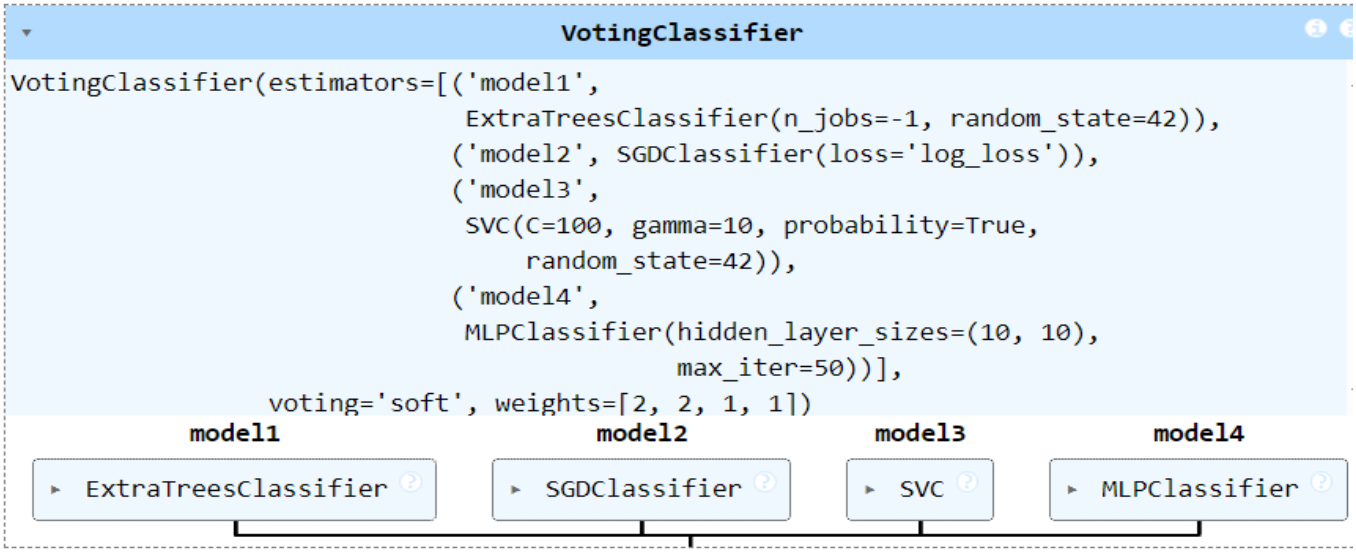

Fig. 4. Snippet of the ensemble classifier from python implementation.

```
Input: unbalanced dataset D={(x1,y1),(x2,y2),……..,(xn,yn)} when yi=[0,1,….k] and k= number of classes
Output: Ensemble Classifier
Cumulative Feature Importance Score: CFISthreshold=[85%,90% 95%,99%]

        Step 1:Preprocessing
                                Step 1a: for each categorical Features Fi in D do
Fi<-- Labelling
end for
        Step 1b: Remove the duplicate instances
        Step 1c: Eliminate Feature Fi with statistical measure 0
        Step 1d: Divide dataset into train(Dtrain) and test(Dtest) set

Step 2: BM<- {SMOTE, RUS, RUS+ROS, RUS+SMOTE}
                        Define the Sampling strategy
                                Select Y min<- minority class
for each BMi generate synthetic instances of Y min as balance dataset DBtrain,DBtest

        Step 3:
Step3a: impscore<-- calculateScore(feature_list,RF, DBtrain, DBtest)
                                Step 3b: imp_feature_list<-- selectFeature(feature_list, impscore,CFISthreshold)
                                Step 3c: get reduced features DBRtrain, DBRtest

        Step 4: voting= ”soft”
Step 4a: selected Fi in DB do
M1=ET(DBRtrain, DBRtestTrain_label)
M2=SGD(DBRtrain, DBRtest ,Train_label)
M3=SVM (DBRtrain, DBRtest ,Train_label)
M4=MLP(DBRtrain, DBRtest ,Train_label)
M5=KNN(DBRtrain, DBRtest ,Train_label)
M6=XGBoost(DBRtrain, DBRtest ,Train_label)

Step 4b: Ensemble_Modeli(DBRtrain, DBRtest,Train_label)
soft_voting_classifier=Concatenate(M1,M2,M3,M4)
        soft_voting_classifier.fit(DBRtrain,Train_label)
predictions=soft_voting_classifier.predict(DBRtest)

Step 5: Result
```

Fig. 5. Pseudocode of the proposed methodology.

IV. Results and Discussion

In this section, the experimental results are analyzed, and the pros and cons of the proposed ensemble models based on the widely used classification metrics are given. The metrics are Accuracy, Precision, Recall, F-Measure and Time complexity. Besides that, another evaluation metric, AUC-ROC, is also chosen for measuring ensemble model classification performance. The EM1, EM2 and EM3 models' evaluation results are presented in Table I, Table II and Table III, respectively. They are obtained from experiments conducted on the CICIDS-2017 benchmark dataset after applying four types of imbalance treatments with random forest feature selection.

Based on these results, the ensemble model consistently achieves 99.88% accuracy across all thresholds (85%, 90%, 95%, 99%). A strong balance between robustness and quality is evident in the precision of 99.67%, recall of 99.58%, and F-Measure of 99.62 %. In terms of time complexity, this model has a low range of 299.572 to 362.86, making it very efficient. By combining these two classifiers, patterns in the data are captured effectively while remaining computationally efficient. As it exhibits high accuracy and reliability at all thresholds, it is an ideal choice for tasks that require precision and speed at the same time. The consistent performance across different thresholds suggests that the model is stable and does not overfit. In spite of its low time complexity, it is scalable for large datasets due to its low time complexity. Overall, this ensemble is a top-performing model and well-suited for real-time applications.

In comparison to the above scenario, this ensemble model delivers slightly lower accuracy, ranging from 99.4% to 99.51%. In this case, precision remained around 99.18%, but recall dropped to 93.52, resulting in a lower F-Measure of approximately 95.74 to 95.79%. The primary downside of this model is its high time complexity at the 90% threshold, which spikes to 2128.69, making it impractical for real-time applications. In contrast, as the threshold increases, the time complexity drops significantly, going from 2128.692 to 555.78 for higher thresholds. This model is uneconomical in comparison with the first scenario because it has a high computational cost for the lower thresholds, despite its reasonable accuracy and F-measure. Small datasets or non-real-time tasks are suitable for this ensemble when time complexity is less of an issue. Overall, the program demonstrates good prediction capabilities, but it doesn't perform well in terms of efficiency.

TABLE I. Effect of Various Performance Metrics for EM1 Model

CFIS	Accuracy (%)	Precision (%)	Recall (%)	F-Measure (%)	Time complexity(sec.)
85%	99.88	99.67	99.58	99.62	299.572
90%	99.88	99.67	99.58	99.62	313.63
95%	99.88	99.67	99.58	99.62	362.86
99%	99.88	9967	9958	99.62	321.58

TABLE II. EFFECT OF VARIOUS PERFORMANCE METRICS FOR EM2 MODEL

CFIS	Accuracy (%)	Precision (%)	Recall (%)	F-Measure (%)	Time complexity(sec.)
85%	99.51	99.18	93.51	95.79	571.62
90%	99.4	99.09	93.51	95.74	2128.692
95%	99.51	99.04	93.51	95.77	499.52
99%	99.43	99.14	93.51	95.77	555.78

TABLE III. EFFECT OF VARIOUS PERFORMANCE METRICS FOR EM3 MODEL

CFIS	Accuracy (%)	Precision (%)	Recall (%)	F-Measure (%)	Time complexity(sec.)
85%	99.6	99.6	99.6	99.6	363.77
90%	99.51	99.51	99.51	99.51	675.66
95%	99.6	99.6	99.6	99.6	388.81
99%	99.6	99.6	99.6	99.6	380.62

All the parameters of this ensemble remain at 99.6% accuracy, precision, recall, and F-Measure. Compared to the first ensemble, this ensemble has a slightly higher level of time complexity between 363.77 and 675.66, which is still manageable. Stability of the model is evident from its identical metrics across thresholds, which indicates strong generalization and robustness. The time complexity of its ensemble is higher than that of the Extra Tree ensemble, but it remains far more efficient than the KNN ensemble in terms of computational efficiency. The model is a great option for applications that require consistency and balance in performance. It is well-suited for tasks involving high-dimensional data or requiring consistent outcomes. Overall, the accuracy and computing efficiency of this ensemble are well balanced.

It can be observed that the three proposed ensemble models yield the same level of performance without any significant differences. This is true for all CFIS scores. For this reason, an 85% CFIS threshold is chosen to evaluate the AUC-ROC metric and then compared with the same CFIS threshold of results in [18]. The relevant confusion matrices are given below.

There are some noticeable improvements in terms of the classification of class 0 and class 2 instances in the second matrix compared to the first matrix, which is derived from the methodology of [18], where they were misclassified. This may result in a potential model with better separation of the model.

The following analysis is grounded on Fig. 6 and Fig. 7. The performance of the EM1 model on class-0 is exceptionally good, and the rate of misclassification is negligible with respect to the size of the actual class 0 compared to the model [18] performance, where there is a high misclassification. The ensemble model reduces the misclassification samples to 50% in the ensemble model for class-1. Accordingly, class-2, class-3 and class-4 performance is very effective with only 37,11 and 14 number of samples being misclassified, which is a negligible percentage compared to the actual number of samples 69037,1650 and 1739, respectively. Coming to class-5, which is the minor class among all, has classified 100% correctly, whereas it is not in line with the first matrix. Fig. 6 shows that there are significant misclassifications in classes 0 and 2.

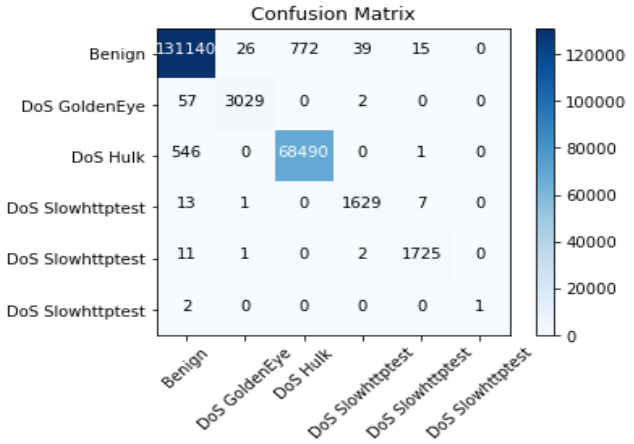

Fig. 6. Confusion Matrix of [18] with CFIS threshold of 85%,

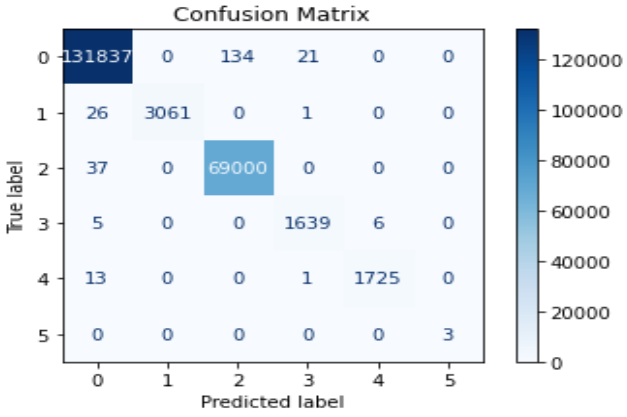

Fig. 7. Confusion matrix of EM1 with CFIS threshold of 85%.

Fig. 8 demonstrates a classifier's performance among six classes in a multi-class Receiver Operating Characteristic (ROC) curve plot. The True Positive Rate (TPR) is represented by the y-axis, and the False Positive Rate (FPR) by the x-axis. Every colored curve represents a distinct class; when all curves reach the upper-left corner, perfect categorization is indicated. Random prediction is represented

by the diagonal dashed line, but the model performs far better. Each class's Area Under the Curve (AUC) is 1.00, indicating perfect performance devoid of misclassifications. These findings point to a remarkable model, but they may also point to possible overfitting, especially if the dataset is tiny or not typical of real-world situations.

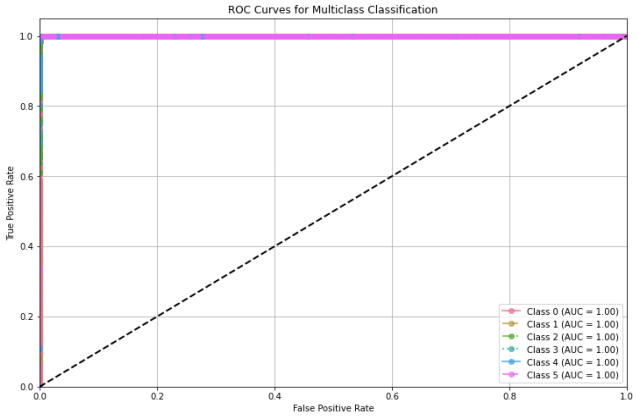

Fig. 8. ROC-AUC curve ensemble model in CICID-2017.

V. Conclusion and Future Scope

In this study, to improve and enhance the performance of NIDS, Ensemble learning is adopted. This technique is used to develop three Ensemble models that are designed in the proposed work. All these models have boosted the performance of the existing method. Each of the proposed Ensemble models has used four classifiers out of six base classifiers, whereas [18] has adopted an SVM binary classifier. It has shown better performance with respect to all the metrics for all chosen re-sampling methods with different CFIS scores. From the analysis of the results of the proposed work, one can observe that an 85% threshold feature subset with a minimum number of features is sufficient for detecting attack classification with minimum computational complexity and without any loss in accuracy. From this process of evaluation, the EM1 model was evaluated with a minimum computational time and gives a better trade-off between computational time and classification metrics. It has been proven that the four re-sampling methods with the EM1 model, along with Random Forest feature selection, outperform. Thus, it is a suggested model for attack detection in real-time, imbalanced traffic of NIDS. The limitation of this study is that the experiments were conducted on a training and testing split only, not executed using cross-validation.

To make the results much clearer, explainable models like SHAPE, LIME can be utilized. As a future scope of this study, statistically based feature selection methods may be chosen for the evolutionary process with different combinations of re-sampling methods or base classifiers. Further, this method can be applied to contemporary datasets for new type of attacks.

References

[1] Y. Sanjalawe and T. Althobaiti, DDoS attack detection in cloud computing based on ensemble feature selection and deep learning, Comput. Mater. Contin., vol. 75, no. 2, pp. 3571–3588, 2023.

[2] M. Ouhssini, K. Afdel, M. Akouhar, E. Agherrabi, and A. Abarda, "Advancements in detecting, preventing, and mitigating DDoS attacks in cloud environments: A comprehensive systematic review of state-of-the-art approaches," Egypt. Inform. J., vol. 27, no. 100517, p. 100517, 2024.

[3] G. S. Kushwah and V. Ranga, "Voting extreme learning machine based distributed denial of service attack detection in cloud computing," J. Inf. Secur. Appl., vol. 53, no. 102532, p. 102532, 2020.

[4] S. Pahal and A. Saroha, "Distributed Denial of Services attacks on cloud servers: Detection, Analysis, and Mitigation," Mapana Journal of Sciences, vol. 22, no. 1, pp. 121–145, 2023.

[5] G. S. Kushwah and V. Ranga, "Optimized extreme learning machine for detecting DDoS attacks in cloud computing," Comput. Secur., vol. 105, no. 102260, p. 102260, 2021.

[6] D. Zeng, J. Zhang, L. Gu, S. Guo, and J. Luo, "Energy-efficient coordinated multipoint scheduling in green cloud radio access network," IEEE Trans. Veh. Technol., vol. 67, no. 10, pp. 9922–9930, 2018.

[7] Y. Mehmood, M. A. Shibli, U. Habiba, and R. Masood, "Intrusion Detection System in Cloud Computing: Challenges and opportunities," in 2013 2nd National Conference on Information Assurance (NCIA), 2013.

[8] S. R. K. Tummalapalli and A. S. N. Chakravarthy, "Intrusion detection system for cloud forensics using bayesian fuzzy clustering and optimization based SVNN," Evol. Intell., vol. 14, no. 2, pp. 699–709, 2021.

[9] F. Palmieri, S. Ricciardi, U. Fiore, M. Ficco, and A. Castiglione, "Energy-oriented denial of service attacks: an emerging menace for large cloud infrastructures," J. Supercomput., vol. 71, no. 5, pp. 1620–1641, 2015.

[10] Y. Li, H. Guo, Q. Zhang, M. Gu, and J. Yang, "Imbalanced text sentiment classification using universal and domain-specific knowledge," Knowl. Based Syst., vol. 160, pp. 1–15, 2018.

[11] R. Panigrahi and S. Borah, "Dual-stage intrusion detection for class imbalance scenarios," Comput. Fraud Secur., vol. 2019, no. 12, pp. 12–19, 2019.

[12] L. Wang and C. Wu, "Dynamic imbalanced business credit evaluation based on Learn++ with sliding time window and weight sampling and FCM with multiple kernels," Inf. Sci. (Ny), vol. 520, pp. 305–323, 2020.

[13] Y. Liu, Z. Yu, C. Chen, Y. Han, and B. Yu, "Prediction of protein crotonylation sites through LightGBM classifier based on SMOTE and elastic net," Anal. Biochem., vol. 609, no. 113903, p. 113903, 2020.

[14] M. E. El-Telbany, "Prediction of the electrical load for Egyptian energy management systems: Deep learning approach," in Advances in Intelligent Systems and Computing, Cham: Springer International Publishing, 2020, pp. 237–246.

[15] Privacy, and Anonymity in Computation, Communication, and Storage: 9th International Conference. SpaCCS; Zhangjiajie, China, 2016.

[16] S. Oyucu, O. Polat, M. Türkoğlu, H. Polat, A. Aksöz, and M. T. Ağdaş, "Ensemble Learning framework for DDoS detection in SDN-based SCADA systems," Sensors (Basel), vol. 24, no. 1, 2023.

[17] K. Alluraiah and M. S. R. Chetty, "Ensemble learning method for DDOS attack mitigation in web based networks," Int. J. Intell. Syst. Appl. Eng., vol. 12, no. 3, pp. 743–753, 2024.

[18] S. Kudithipudi, N. Narisetty, G. R. Kancherla, and B. Bobba, "Evaluating the efficacy of resampling techniques in addressing class imbalance for network intrusion detection systems using support vector machines," Ing. Syst. D Inf., vol. 28, no. 5, pp. 1229–1236, 2023.

[19] K. Swarnalatha, N. Narisetty, G. Rao Kancherla, and B. Bobba, "Analyzing resampling techniques for addressing the class imbalance in NIDS using SVM with Random Forest feature selection," International Journal of Experimental Research and Review, vol. 43, pp. 42–55, 2024.

[20] E. A. Al-Qarni and G. A. Al-Asmari, "Addressing imbalanced data in network intrusion detection: A review and survey," Int. J. Adv.

Comput. Sci. Appl., vol. 15, no. 2, 2024.

[21] J. Mijalkovic and A. Spognardi, "Reducing the false negative rate in deep learning based network Intrusion Detection Systems," Algorithms, vol. 15, no. 8, p. 258, 2022.

[22] H. Zhang, B. Zhang, L. Huang, Z. Zhang, and H. Huang, "An efficient two-stage network intrusion detection system in the internet of Things," Information (Basel), vol. 14, no. 2, p. 77, 2023.

[23] N. D. Patel, B. M. Mehtre, and R. Wankar, "A computationally efficient dimensionality reduction and attack classification approach for network intrusion detection," Int. J. Inf. Secur., vol. 23, no. 3, pp. 2457–2487, 2024.

[24] D. Cemernek, S. Siddiqi, and R. Kern, Effects of Class Imbalance Countermeasures on Interpretability. IEEE Access, 2024.

[25] A. A. Abdulrahman and M. K. Ibrahem, "Toward constructing a balanced intrusion detection dataset based on CICIDS2017," Samarra Journal of Pure and Applied Science, vol. 2, no. 3, pp. 132–142, 2020.

[26] P. Barnard, N. Marchetti, and L. A. DaSilva, "Robust network intrusion detection through explainable artificial intelligence (XAI)," IEEE Netw. Lett., vol. 4, no. 3, pp. 167–171, 2022.

[27] N. Tran, H. Chen, J. Jiang, J. Bhuyan, and J. Ding, "Effect of class imbalance on the performance of machine learning-based network intrusion detection," International Journal of Performability Engineering, vol. 17, no. 9, 2021.

[28] O. Mjahed, S. El Hadaj, E. Mahdi El Guarmah, and S. Mjahed, "New denial of service attacks detection approach using hybridized deep neural networks and balanced datasets," Comput. Syst. Sci. Eng., vol. 47, no. 1, pp. 757–775, 2023.

[29] A. Abdelkhalek and M. Mashaly, "Addressing the class imbalance problem in network intrusion detection systems using data resampling and deep learning," J. Supercomput., vol. 79, no. 10, pp. 10611–10644, 2023.

[30] M.-G. Kim and H. Kim, "Anomaly detection in imbalanced encrypted traffic with few packet metadata-based feature extraction," Comput. Model. Eng. Sci., vol. 141, no. 1, pp. 585–607, 2024.

[31] A. S. Barkah, S. R. Selamat, Z. Z. Abidin, and R. Wahyudi, "Data Generative Model to Detect the Anomalies for IDS Imbalance CICIDS2017 Dataset," Dataset. TEM Journal, no. 1, 2017.

[32] C. Wang, D. Xu, Z. Li, and D. Niyato, "Effective intrusion detection in highly imbalanced IoT networks with lightweight S2CGAN-IDS," arXiv [cs.CR], 2023.

[33] M. Mbow, H. Koide, and K. Sakurai, "Handling class imbalance problem in intrusion detection system based on deep learning," International Journal of Networking and Computing, vol. 12, no. 2, pp. 467–492, 2022.

[34] A. S. Barkah, S. R. Selamat, Z. Z. Abidin, and R. Wahyudi, "Impact of data balancing and feature selection on machine learning-based network intrusion detection," JOIV Int. J. Inform. Vis., vol. 7, no. 1, p. 241, 2023.

[35] P. Goyal, R. Rani, and K. Singh, "Comparative analysis of machine learning and ensemble learning classifiers for Alzheimer's disease detection," in 2022 IEEE International Conference on Current Development in Engineering and Technology (CCET), 2022.

[36] H. M. Saleh and A. Hend Marouane, "Stochastic gradient descent intrusions detection for wireless sensor network attack detection system using machine learning," IEEE Access, vol. 12, pp. 3825–3836, 2024.

[37] N. Nirmalajyothi, K. Rao, B. B. Rao, and K. Swathi, "Performance of Various SVM Kernels for Intrusion Detection of Cloud Environment," International Journal of Emerging Trends in Engineering Research, vol. 8, no. 10, 2020.

[38] A. A. Hagar and B. W. Gawali, "Implementation of machine and deep learning algorithms for intrusion detection system," in Intelligent Communication Technologies and Virtual Mobile Networks, Singapore: Springer Nature Singapore, 2023, pp. 1–20.

[39] S. Džeroski and B. Ženko, "Stacking with multi-response model trees," in Lecture Notes in Computer Science, Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, pp. 201–211.

[40] I. Gandhi and M. Pandey, "Hybrid Ensemble of classifiers using voting," in 2015 International Conference on Green Computing and Internet of Things (ICGCIoT), 2015.

[41] S. W. A. Sherazi, J.-W. Bae, and J. Y. Lee, "A soft voting ensemble classifier for early prediction and diagnosis of occurrences of major adverse cardiovascular events for STEMI and NSTEMI during 2-year follow-up in patients with acute coronary syndrome," PLoS One, vol. 16, no. 6, p. e0249338, 2021.