DOI: https://doi.org/10.52756/ijerr.2024.v43spl.004 *Int. J. Exp. Res. Rev.*, Vol. 43: 42-55 (2024)

Original Article | Peer Reviewed | Open Access

International Journal of Experimental Research and Review (IJERR)

ISSN: 2455-4855 (Online)
www.iaph.in
ISSN 2455-4855

# Analyzing Resampling Techniques for Addressing the Class Imbalance in NIDS using SVM with Random Forest Feature Selection

**K. Swarnalatha[1], Nirmalajyothi Narisetty[2]*, Gangadhara Rao Kancherla[1] and Basaveswararao Bobba[1]**

[1]Department of Computer Science and Engineering, Acharya Nagarjuna University, Guntur, Andhra Pradesh-522510, India; [2]Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Bowrampet, Hyderabad, Telangana-500090, India

**E-mail/Orcid Id:**

*KS,* latha80@gmail.com, https://orcid.org/0009-0005-4325-6254; *NN,* nirmala1.narisetty@gmail.com, https://orcid.org/0000-0002-4810-9676; *GRK,* kancherla123@gmail.com, https://orcid.org/0000-0002-6106-8477; *BB,* bobbabrao62@gmail.com, https://orcid.org/0000-0003-4287-0891

**Article History:**

**Received:** 19th Jun., 2024
**Accepted:** 17th Sep., 2024
**Published:** 30th Sep., 2024

**Keywords:**
Cloud Computing, Cumulative Feature Importance, Intrusion Detection System, Machine Learning, Resampling Methods

**How to cite this Article:**
K. Swarnalatha, Nirmalajyothi Narisetty, Gangadhara Rao Kancherla and Basaveswararao Bobba (2024). Analyzing Resampling Techniques for Addressing the Class Imbalance in NIDS using SVM with Random Forest Feature Selection. International Journal of Experimental Research and Review, 43, 42-55.
**DOI:**
https://doi.org/10.52756/ijerr.2024.v43spl.004

**Abstract:** The purpose of Network Intrusion Detection Systems (NIDS) is to ensure and protect computer networks from harmful actions. A major concern in NIDS development is the class imbalance problem, i.e., normal traffic dominates the communication data plane more than intrusion attempts. Such a state of affairs can pose certain hazards to the effectiveness of detection algorithms, including those useful for detecting less frequent but still highly dangerous intrusions. This paper aims to utilize resampling techniques to tackle this problem of class imbalance in NIDS using a Support Vector Machine (SVM) classifier alongside utilizing features selected by Random Forest to improve the feature subset selection process. The analysis highlights the combativeness of each sampling method, offering insights into their efficiency and practicality for real-world applications. Four resampling techniques are analyzed. Such techniques include Synthetic Minority Over-sampling Technique (SMOTE), Random Under-sampling (RUS), Random Over-sampling (ROS) and SMOTE with two different combinations i.e., RUS SMOTE and RUS ROS. Feature selection was done using Random Forest, which was improved by Bayesian methods to create subsets of features with feature rankings determined by Cumulative Feature Importance Score (CFIS). The CIDDS-2017 dataset is used for the performance evaluation, and the metrics used include accuracy, precision, recall, F-measure and CPU time. The algorithm that performs best overall in the CFIS feature subsets is SMOTE, and the features that give the best result are selected at the 90% level with 25 features. This subset accomplishes a relative accuracy enhancement of 0.08% than the other approaches. The RUS+ROS technique is also fine but somehow slower than SMOTE. On the other hand, RUS+SMOTE shows relatively poor results although it consumes less time in terms of computational time compared to other methods, giving about 50% of the performance shown by the other methods. This paper's novelty is adapting the RUS method as a standalone test for screening new and potentially contaminated datasets. The standalone RUS method is more efficient in terms of computations; the algorithm returned the best result of 98.13% accuracy at 85% at the CFIS level of 34 features with a computation time of 137.812 s. It is also noted that SMOTE is considered to be proficient among all resampling techniques used for handling the problem of class imbalance in NIDS, vice 90% CFIS feature subset. Future research directions could include using these techniques in different data sets and other machine learning and deep learning methods together with ROC curve analysis to provide useful pointers to NIDS designers on how to select the right data mining tools and strategies for their projects.

## Introduction

The rapid evolution of communications and emerging technologies make use of cloud computing techniques to access various internet-based applications. The cloud computing industry has been one of the fastest-growing segments of IT over the past few years due to its promising business model and fast growth.

***Corresponding Author:** nirmala1.narisetty@gmail.com

With cloud computing, IT infrastructure management and maintenance costs are reduced. Cloud resources are controlled over the internet by different organizations using standards and protocols (Kumar et al., 2013; Akgun et al., 2022). This makes IT infrastructure more susceptible to attacks due to its distributed nature and centralized control. Thus, it faces some challenges.

A major obstacle to the success and adoption of cloud computing by organizations or any individuals is its security (Elmasry et al., 2021). Among the most popular intrusions in clouds are attacks such as Distributed Denial of Service (DDoS), which result in service degradation or a denial of service. In a DDoS attack, unauthorized endeavours aim to interrupt the services of a networked host temporarily or permanently to prevent it from being used by its safeguards against such attacks into Cloud services to facilitate widespread cloud adoption. To address these anomalous networks, the network security intended users. Ultimately, a DDoS attack may result in losing the customers' trust, operational impact, and reputation brand of a particular product (Huhn, 2021). So, it would be beneficial to integrate researchers who implemented various NIDS (Salo et al., 2019).

There have been some successes and advancements in intrusion detection thanks to ML and DL, but many obstacles remain to overcome (Zhang et al., 2020). Most intrusion detection systems still struggle with class-imbalanced information. An imbalance has occurred when the number of official user trials is significantly higher than the number of invader trials. Nevertheless, the available network intrusion datasets are imbalanced with different types of minority attack samples. Detecting these minority attacks is becoming a challenge in intrusion detection and affecting the performance of IDS. And also, the importance of features plays a vital role in balancing these samples. At present, an ideal technique is introduced to enhance the performance of IDS by extracting the important features and balancing the samples with the adoption of various data-level techniques. Machine learning and deep learning have made strides in intrusion detection, achieving positive results. However, this progress isn't without its hurdles. Significant challenges still need to be addressed to fully leverage these advancements for robust network security (Narisetty et al., 2021; Narisetty et al., 2021). One major hurdle is a class imbalance, where legitimate user traffic vastly outnumbers malicious activity. This imbalance hinders the performance of most Intrusion Detection Systems (IDS) because the models become biased towards the majority class (legitimate traffic) and struggle to accurately detect the minority class (attacks). Nevertheless, the available network intrusion datasets are imbalanced with different types of minority attack samples (Nayani et al., 2021). Detecting these minority attacks is becoming a challenge in intrusion detection and affecting the performance of IDS. Also, features play a vital role in balancing these samples (Rao et al., 2021; Madhuri et al., 2022). At present, an ideal technique has been introduced to improve the performance of IDS by extracting the important features and balancing the samples with the adoption of various data-level techniques.

The study involves a comprehensive analysis of the impact of different resampling techniques by addressing the class imbalance issue through effective resampling techniques and leveraging advanced feature selection methods and this research aims to contribute to the development of more robust and accurate Network Intrusion Detection Systems.

i. To investigate the performance impact of the resampling methods and the combination of these methods for addressing the class imbalance problem through the SVM classifier when imposing the Random Forest feature selection approach.
ii. An SVM classifier is utilized with Random Forest feature selection to achieve this goal. Bayesian Optimization (BO) with a tree-based Parzen estimator is employed to identify optimal feature subsets based on cumulative feature importance score thresholds. The binary classification evaluation is conducted on the CICIDS-2017 dataset.
iii. Different numbers of feature sets are formed, based on cumulative feature importance score criteria. Then, the performance measures of these sets will be computed with computational time through experiment evaluation.

The study culminates in offering practical suggestions to network administrators on implementing effective mitigation algorithms in Network Intrusion Detection Systems (NIDS). These recommendations are based on the comprehensive analysis of resampling techniques and their impact on performance and computational efficiency.

The rest of this paper is well-organized as follows: Section 2 discusses the state of the art concerning the intrusion detection models in conjunction with recent datasets using statistical and ML techniques. A detailed description of the intrusion detection framework and its framework is presented in Section 3. Followed by the experimental results of the suggested approach are discussed. Lastly, the pertinent conclusion and provided future research directions.

DOI: https://doi.org/10.52756/ijerr.2024.v43spl.004

## Literature Review

There has been a lot of study into intrusion detection systems (IDS) over the past 20 years, with a focus on how to best use Machine Learning approaches in conjunction with different feature selection and class balance algorithms. This section aims to offer a synopsis of relevant work and the contributions made by those involved.

Awad et al. (2019) aimed to address the shortcomings of prior work and the dangers presented by small classes to boost IDS performance. Stratified sampling is used to choose a subset of the data in a way that takes into account all classes and their ratios. Next, the hidden layer's weights and biases are randomly selected to train the model using an Extreme Learning Machine. The primary experiments were carried out using the UNB ISCX2012 dataset. Compared to other models, ELM models using polynomial functions performed better in terms of accuracy, recall, and F-score. The Normal, DoS, and SSH categories were also where it faced off against more conventional models. Nevertheless, computational complexity was not the author's primary focus.

Mbow et al. (2022) suggested a hybrid model that combines the over-sampling SMOTE and under-sampling Tomek link techniques to address the imbalance problem in datasets. Next, we put the aforementioned techniques to the test using DL models like LSTM and CNN. Then, three separate datasets—NSL-KDD, CICIDS2017, and CICIDS2018—are used to evaluate the suggested methodology. The model is subjected to 10-fold cross-validation to make it more generalizable. It is compared to several of the baseline models' current approaches to prove the suggested method's merits and utility in addressing the imbalanced issue. Results show that intrusion detection efficiency has been improved, leading to a lower false alert rate. However, the intrusion detection rate is low compared to the present study.

Wang et al. (2023) proposed a random forest classifier is used to identify if a sample is an assault or not. The next step is to build an autoencoder (AE) using only safe training set samples. Experiments simulating an unknown assault involve removing data related to a specific attack category from the training set. Two hyperparameters, RF probability and MSE, have been employed in the detection technique. This paper's findings indicate that the second step successfully classifies the samples that were initially misclassified. Two datasets, NF-CSE-CIC-IDS2018-v2 and NF-BoT-IoT-v2, are used to assess this method. Finally, the experimental findings showed that combining RF and AE increased the detection rate while decreasing FPR compared to single detection approaches.

Alqarni and El-Alfy, Proposed an Intrusion Detection model (Wang et al., 2023) TrafficImbalanceNet is based on generative deep learning to address traffic imbalances in networks. They trained a Conditional Tabular Generative Adversarial Network (CTGAN) on the over-sampled minority class examples to balance out the dataset. We experimented using the SVM, KNN and DT classifiers to determine their effectiveness when trained on the skewed NSL-KDD dataset. The results of unbalanced learning for intrusion detection were as follows: CTGAN could substantially enhance the performances of both SVM and DT. KNN does not need resampling because it is robust when dealing with class imbalance, so its performance was the same. Furthermore, the study showed that CTGAN is more efficient in modelling discrete-feature distribution than continuous ones. However, the authors did not consider the complexity of the time in this study.

The article by Mohammad et al. (2021) proposed a hybrid feature selection model to detect abnormal activities in computer systems. Where the model combines two optimization techniques Grey Wolf Optimization (GWO) and Particle Swarm Optimization (PSO), well yes you read me correctly, it picks bits of both to create the best-performing subset. Then, the model was evaluated using artificial neural networks (ANN) and naive Bayes (NB) using the UNSW-NB15 data set. It can be observed that the features for intrusion detection are extensively selected by PSO and GWO algorithms. PSO and GWO extract the features and then concatenate to get a promising result with the fewest no. of features. The research compared the PSO-GWO-ANN and Psoriasis classifiers in terms of hybrid classifier based on feature selection and intrusion detection.

Zekan et al. (2022) proposed a new semi-supervised EC-GAN approach for network flow in the domain of NIDS. The researchers modified the original EC-GAN to use tabular data, specifically applied to CIC-IDS-2017 dataset. This work shows that the game theoretical framework along with synthetic data generation and deep neural network classifier used in EC-GAN, succeeded in addressing the challenges of false positive rate as well as detection rate on low sample imbalanced datasets. In summary, the results indicate that using synthetic data combined with EC-GAN improves how an IDS can perform. Using only 25% of the original dataset, our EC-GAN classifier outperforms state-of-the-art alternatives and attains an excellent F1 score of 0.9995 whilst maintaining a low false positive rate of about 0.0005.

Babu et al. (2023) in their work focused on a method called modified conditional generative adversarial

DOI: https://doi.org/10.52756/ijerr.2024.v43spl.004

network (MCGAN), which deals with imbalanced class problems in intrusion detection. The MCGAN generates contain-specific samples targeting a balanced population of the majority and minority classes to alleviate the negative impact induced by class imbalance. The method does extensive intrusion categorization by integrating MCGAN with Bi-LSTM (Bidirectional Long Short-Term Memory). In absolute terms, the suggested method achieves an accuracy of 95.16% in experiments with dataset NSL-KDD+ using 20 selected features. When following our method to the NSL-KDD+ datasets selectively using only 20 features, we achieved good performance in terms of accuracy, precision and FPR with no degradation for actual detection rate (F1-score).

In their work, Zhang et al. (2020) proposed a network intrusion detection system (CWGAN-CSSAE ) that solves the two most important issues, very rare attacks with no data and unknown attacks. This work integrates two methods: a more advanced generative adversarial network (CWGAN) to provide extra data for rare attacks and the stacked autoencoders (CSSAE) are used to extract important features from raw network traffic. CWGAN-CSSAE achieves an accuracy of over 90% on multiple datasets by improving the detection performance for rare and potentially unknown attacks. This is possible for strong network security.

Al et al. (2021) presented a new classification-based attack detection system using big data analytics tools for real-time processing of enormous raw traffic. Based on deep learning to enhance its intrusion detection capability. It specifically uses a Hybrid Deep Learning network, which integrates the topologies of Convolutional Neural Networks with Long Short-Term Memory. The local SAM-TLM uses only SMOTE (Synthetic Minority Over-sampling Technique), and the Tomek link combines both under the umbrella term as an STL approach to alleviate the robustness of NIDS accuracy on data imbalance. We use the CIDDS-001 data set for evaluations, with more classes and the UNS-NB15 dataset when only five binary classes. To demonstrate the proposed method, we evaluated the proposed method against nine different ML and DL algorithms in research. Some of the important performance metrics used to assess your findings are accuracy, F-measure, recall, ROC curves, and precision-recall curve. The experimental data show that the suggested method's multiclass and binary classifications are accurate enough. I also want to highlight its multiclass testing accuracy of 99.83% and the binary recognition test with a final overall score on that label classification, scoring 99.17%. Such results support the claim that our proposition works better on network assault detection with imbalanced datasets than existing state-of-the-art algorithms.

Hagar and Gawali (2022) compared the performance of a deep learning model and one machine learning model in the intrusion detection system. Four feature extraction techniques were applied before putting it into these algorithms to generate an effective dataset. For all feature extraction techniques, random forest offers better results than MLP. A maximum accuracy of 99.90% was achieved with 36 features and a false positive rate (FPR) of 0.068%. The results show that an MLP algorithm in DL and RF algorithm in ML have increased accuracy and decreased FPR.

Researchers Jovana Mijalkovic and Angelo Spognardi (Mijalkovic et al., 2022) aimed to address "high false negative rates and low predictability for minority classes" in their work. There are three steps in the proposed methodology: correcting training and testing subset distributions, selecting features, and applying class weights. The proposed methodology is evaluated on the NSL KDD and UNSW-NB15 datasets. Results indicate that careful selection of parameters can effectively trade-off between FNR, accuracy, and minority class detection.

Existing Intrusion Detection Systems (IDS) struggle to identify DoS/DDoS attacks with traditional Machine Learning methods. The study (Mjahed et al., 2023) addresses this issue by proposing a novel DNN-based approach that incorporates Metaheuristic algorithms for improved accuracy. It Utilizes real-world imbalanced datasets (CICIDS 2017-2019) with normal and DoS/DDoS attack data. Pre-processes data using K-means balancing and feature selection with LDA to enhance DNN performance. Introduces a unique combination of four Metaheuristic algorithms (AIS, FA, IWO, CS) with DNNs. Achieves exceptional accuracy (over 99.9%) in DoS/DDoS attack detection.

Security concerns in the growing IoT landscape are addressed by Zhang et al. (2023). A two-stage intrusion detection model is proposed. It leverages both ML and DL for efficient and fine-grained attack detection on large datasets (using CSE-CIC-IDS2018). Stage 1 employs machine learning, with LightGBM achieving high accuracy (99.135%) and fast training in identifying benign and abnormal traffic patterns. In Stage 2, the Convolutional Neural Network (CNN) is utilized to classify the anomalies identified in Stage 1 in detail. This stage demonstrates excellent performance (over 99.8% accuracy) even with imbalanced datasets. The model achieves high efficiency with a total training time of 74.8 seconds, surpassing existing methods in handling large-scale data.

DOI: https://doi.org/10.52756/ijerr.2024.v43spl.004

The research methodology proposed by Chui et al. (2023) tackles the critical issue of imbalanced datasets hindering network intrusion detection in smart cities. The proposed solution is a three-stage data generation algorithm that utilizes various techniques to create high-quality data and balance minority classes. This approach helps overcome bias caused by skewed datasets in current benchmarks. The authors suggested exploring transfer learning, merging diverse datasets, and incorporating different data generation methods. These advancements hold promise for building even more robust training data, ultimately leading to enhanced security.

The research by Gwiazdowicz et al. (2023) outlined key practices for building stronger machine-learning models for threat detection, particularly when dealing with imbalanced datasets (datasets with uneven class distribution). It emphasizes using robust feature selection techniques like random forests to identify the most informative data, leading to faster and more interpretable models. To assess the effectiveness of these methods, including a baseline model for comparison is crucial. Furthermore, relying solely on accuracy metrics can be misleading when evaluating models on imbalanced data. The study suggests incorporating F1-score, Cohen's kappa, and accuracy for a more comprehensive picture. Finally, the importance of transparency and data quality is stressed. Documenting every step of model development allows for replication and future comparisons, while proper data preparation ensures the model's accuracy and reliability. By following these guidelines, researchers can develop more robust and reliable ML models for threat detection.

The GMM-WGAN-IDS multi-module intrusion detection system, introduced by Cui et al. (2023), can manage imbalanced and high-dimensional datasets. It fixes two major issues, feature redundancy and class imbalance, which improves detection performance overall and is particularly useful for unusual attacks. The system utilizes a Stacked Autoencoder (SAE) module for efficient feature extraction. A Generative Adversarial Network with a Gaussian Mixture Model is employed to handle data imbalance. Lastly, a convolutional neural network combined with long short-term memory is employed to correct classification. Results showing increased detection accuracy, especially for infrequent attacks, are encouraging and demonstrate the system's efficiency. To improve performance further, scientists intend to investigate the future use of attention processes and convolutional GANs.

## Materials and Methods

The IDS framework is shown in Figure 1 and has been designed using machine learning for the comparative study of different resampling techniques. Some of the most basic modules of the system are data pre-processing, feature extraction and determination of hyperparameters, and finally classifying the attacks. Because modern attacks on intrusion detection systems that work in cloud environments are dynamic, the developed IDS can successfully be used in various segments of the cloud environment to combat such threats.

### Pre-processing

As the first step in the development of an IDS, sufficient information is collected on network traffic which encompasses both benign and contemporary common attacks, presenting a realistic picture of real-world attacks i.e., CICIDS-2017. The dataset encompasses network traffic over 5 days, beginning at 09:00 on Monday, July 3, and closing at 17:00 on Friday, July 7. The network data was captured with a CIC Flow Meter (2017) while it was sorted into labelled flows using timestamps, IPs, ports, protocols and types of attacks. The dataset is publicly available in the form of packet capture files, otherwise known as PCAP. This study focuses on intrusion detection concerning the Wednesday dataset, which includes 692,703 instances in 80 columns. This dataset has six classes: Benign, DoS GoldenEye, DoS Hulk, DoS Slowhttptest, DoS Slowloris, and DoS Heartbleed (Sharafaldin et al., 2018).

According to the basic statistical details (Narisetty et al., 2021), eight features have zero min and max values, which means that analysing these features will not add any value to the analysis, so those features dropped. When the dataset involves null records Machine Learning (ML) algorithms cannot be built on those records because ML algorithms are mathematical equations, and require a value. In this dataset, the null records associated with each class are not considered since their percentage relevant to the class is small. To improve the algorithm's robustness, a normalization step is performed to equalize the magnitudes of different features, which helps avoid dominance in the learning process.

To improve the algorithm's robustness, the normalization step is performed to equalize the magnitudes of different features, which helps avoid dominance in the learning process. By looking at Table 1 it is observed that the dataset is imbalanced. The fraudulent instances are significantly lower than normal class instances i.e., accounting for around 37% of the total number of observations. The five different attack

DOI: https://doi.org/10.52756/ijerr.2024.v43spl.004

types are distributed into 4%, 91%, 2%, 2% and 0.4% of total attack instances.

A substantial amount of labelled data samples is often needed to guarantee precise detection of intrusion activities in learning-based Intrusion Detection Systems (IDS). The labelled training set's amount and quality greatly influence how well the NIDS performs. The issue of producing high-quality training cases stems from the fact that conventional labelling techniques entail complications and are prone to errors in manual processes. In the event of a data class imbalance, a miss classification characterized by extremely large majority classes and extremely small minority classes may result.

In practical scenarios, the training of machine learning models using imbalance as well as extensive volumes of network traffic data is impractical and can incur significant time costs, particularly during the hyperparameter tuning phase, which necessitates multiple iterations of model training. Data resampling emerges as a prevalent technique to enhance the efficiency of model training, enabling the balancing of classes from the original data and thereby reducing training complexity. The four resampling methods from (Kudithipudi et al., 2023.), namely SMOTE, RUS, RUS+ROS, and RUS+SMOTE are considered to balance the class labels.

## Feature Engineering

A feature selection approach becomes increasingly important, particularly when the dimensions of the dataset are high. For the ML model, generalization capability can be enhanced by removing irrelevant features, as well as the training process can be expedited when the irrelevant features are removed. The importance of features is often used as a criterion for selecting features in ensemble learning methods based on decision trees. The Random Forest has proven to be a valuable algorithm capable of addressing feature selection challenges, even in datasets with a high number of variables. Eliminating unimportant variables enhances both classification accuracy and performance (Chen et al., 2020; Fong et al., 2013).

As an extension of the study (Kudithipudi et al., 2023), SMOTE, RUS, RUS+ROS, and RUS+SMOTE methods for handling imbalanced data were tested with feature selection methods by varying numbers of features. Initially, the importance of individual features in the dataset is calculated by a random forest algorithm with default parameters. Nevertheless, there were not many differences concerning time complexity as well as performance. So, to improve the runtime, and storage space and mitigate the overfitting problem parameter

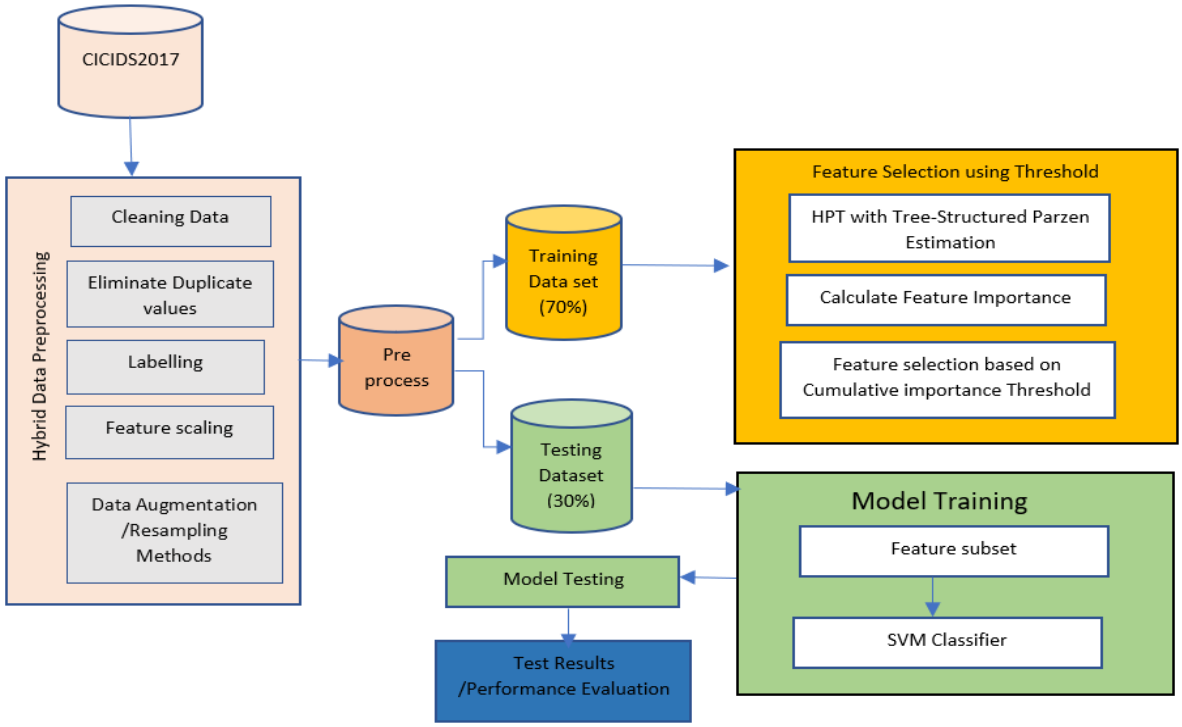


**Figure 1. Proposed Research Methodology.**

DOI: https://doi.org/10.52756/ijerr.2024.v43spl.004

tuning (HP) of random forest is carried out using Bayesian optimization (BO) using a tree-based Parzen estimator. BO (Yang et al., 2020) is a commonly used method for HP optimization problems.

**Table 1. Imbalance ratio of each label in the given dataset.**

| Class labels | No. of instances | Imbalance % w.r.t to majority Class | Imbalance % w.r.t. the total number of instances |
|---|---|---|---|
| Benign | 440,031 | 1 | 0.6352 |
| DoS GoldenEye | 10,293 | 0.0233 | 0.0148 |
| DoS Hulk | 231,073 | 0.5251 | 0.3335 |
| DoS Slowhttptest | 5,499 | 0.0124 | 0.0079 |
| Dos Slowloris | 5,796 | 0.0131 | 0.0079 |
| Heartbleed | 11 | 0.00002 | 0.00001 |

Hyperparameter optimization (HPO) involves using an optimization algorithm to reduce the error rate of a machine learning model for a certain problem or data (Rish, 2001). One of the optimization strategies is Bayesian optimization, which corresponds to a small subset sampled from the HPO. Based on previous evaluation results, it attempts to learn experience and decide which hyperparameter value should be executed next (Sulzmann et al., 2007). To discover the next data point from an objective, a surrogate model covering all of the observed so far is utilized in conjunction with an acquisition function. The BO's two favorite surrogate models are the Gaussian process (GP) and the Tree-Structured Parzen estimation (TSPE).

The second model TSPE is utilized in this work since the random forest is also a tree-based structure. Two density functions, l(x) and g(x), are introduced in BO-TPE and are used to generate all domain variables. TPE classifies observed results into favourable and unfavourable groups based on a predefined percentile y∗. After processing these two sets of results, simple Parzen windows are used to analyze them (Soliman and Mahmoud, 2012).

$$P(x|y,D) = \begin{cases} \boldsymbol{l(x)}, & \boldsymbol{if\ y < y*} \\ \boldsymbol{g(x)} & \boldsymbol{if\ y > y*} \end{cases}$$

Here, the two hyperparameter values, l(x) and g(x) stand for the likelihood of finding each of the areas with high and low performance. BO-TSPE uses the ratio l(x)/g(x) to calculate the ideal hyperparameter values. The Parzen estimators are prearranged in a tree manner, which is significant since it guarantees the preservation of certain conditional relationships between hyperparameters. Furthermore, BO-TPE functions well with a variety of hyperparameters.

The Parzen estimators guarantee the maintenance of certain conditional dependencies by their tree-like topology. Optimization of the hyperparameters of tree-based models, which frequently contain numerous parameters, is accomplished using BO-TPE. The primary parameters, parameter descriptions, tuning parameters, and optimal parameters are shown in Table 2.

The identified best parameters are then used to calculate the importance of each feature in the dataset. The statistics prove that some variables are unimportant and their scores are very low. The insignificant nature of so many variables (or the fact that rounding yields nearly insignificant values) should allow us to remove some of them without adversely affecting the results.

The identified best parameters are then used to calculate the importance of each feature in the dataset. The statistics prove that some variables are unimportant and their scores are very low. The insignificant nature of so many variables should allow us to remove some of them without adversely affecting the results. It is observed that the maximum importance score for feature 57 is 0.091. About 22% of features (16) have importance scores close to zero. A score close to zero means they don't contribute any information to the model learning process. To assess feature importance, we plotted the cumulative feature importance scores. As shown in Figure 3, only 52 features were needed to achieve a cumulative score of 1. This suggests that the remaining features may have minimal impact on the model's performance. Based on these, the top features were selected to implement the IDS; nevertheless, there was no improvement in performance or time complexity. Therefore, the number of features for cumulative importance started to decrease by 99%, 95%, 90%, and 85%, respectively, with a combination of all four resampling methods as discussed above. In total, 16 experiments were conducted to analyse the effect of feature importance and the results are recorded and discussed in the following section.

## Classification

We do the tests on a real-world system with Windows 10, an Intel Core i5 processor running at 1.80 GHz, and 8 GB of RAM. With the help of Python and the scikit-learn framework, the IDS model is put into action. The dataset CICIDS2017 is used to evaluate the assertiveness of resampling methods. The feature importance scores are obtained based on the RFFS method with optimized hyperparameter tuning, as shown in Table 2. The four

DOI: https://doi.org/10.52756/ijerr.2024.v43spl.004

feature subsets are constructed based on cumulative feature importance score thresholds i.e., 85%, 90%, 90%, 95%, and 99 %. To conduct experiments with four Cumulative Feature Importance Score (CFIS) based feature subsets are input to the SVM classifier. The four scenarios are called 85% CFIS, 90% CFIS, 95% CFIS, and 99% CFIS. The following section compares the performance of four resampling methods, with the adaption of two standalone resampling methods (SMOTE, RUS) and two combination methods (RUS+ROS, RUS+SMOTE). The RFFS is applied to these four methods using the SVM classifier. After obtaining the above-mentioned CFIS criteria feature subsets are taken as input data sets. The results of the experiments, including the F-measure, recall, accuracy, and precision concerning the time it took to develop the model, will be detailed in the following session.

**Table 2. The identified Optimal hyperparameters for random forest.**

| Model Parameters | Tuning Values | Optimal Value | |
|---|---|---|---|
| n_estimators | [10,100,200] | 190 | |
| min_samples_leaf | [1,5,11] | 1 | |
| max_depth | [5,25,50] | 27 | |
| max_features | [1,10,20] | 18 | |
| min_samples_split | [2,5,11] | 8 | |
| Criterion | | ['Gini', 'Entropy'] | Gini |

## Results and Discussion

The highlights of the results based on the conducted experiments using the CICIDS2017 dataset are exhibited in the following tables and figures. The results and discussions are presented in the following sub-sections according to four scenarios.

### 85% CFIS Scenario

From the above Table 3 and Figure 4, it is identified that the accuracy of RUS+ROS is high among the other methods, with a value of 99.13%. The RUS and SMOTE are in the next order with a marginal difference, whereas the accuracy of RUS+SMOTE is much less, i.e., 53.565%. The precision, recall, and F-measures also follow the same accuracy directions. In this scenario RUS+SMOTE method is performing very poor results for all the metrics (at around 50%± 5%) except recall, which performs 85%, which is an insignificant difference of 3% less. It indicates that all the methods uniformly behave according to correctly identified as belonging to that class.

### 90% CFIS Scenario

Interpretation of 90% CFIS Scenario results from the above Table 4 and Figure 5 it is noted that the accuracy of SMOTE and RUS+ROS is in higher order among the other methods (99.36% and 99.13%, respectively) with a negligible difference of 0.23%. The RUS is next in order, with a difference of 0.86% from RUS+ROS. It is

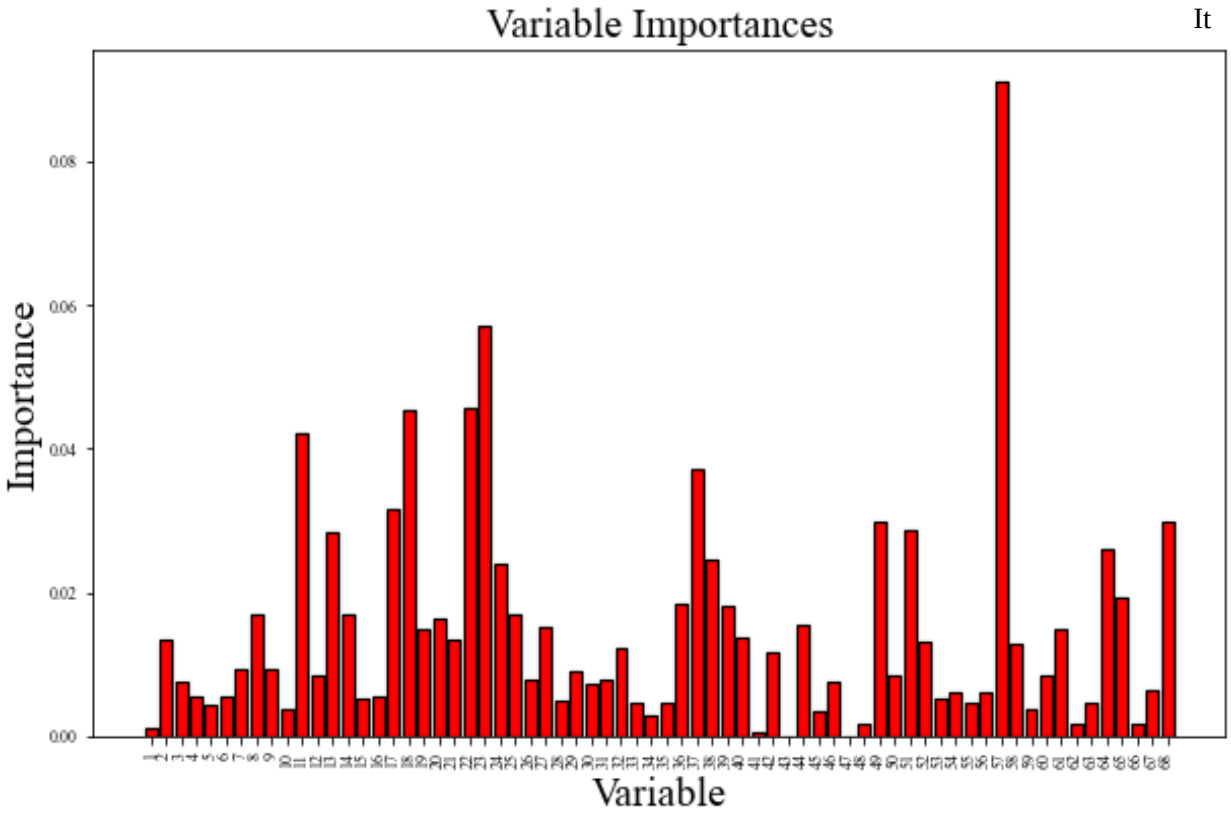


**Figure 2. Visualization of the feature importance in the CICIDS-2017 dataset.**

observed that the precision, recall and F-measure metricsare given in order of SMOTE, RUS and RUS+ROS, but SMOTE gives higher performance than the other two methods.

### 95% CFIS Scenario

Table 5 and Figure 6 show that all four metrics of SMOTE provide the best performance of the other three methods. Subsequently, RUS+ROS and RUS methods exhibit one after other better results for all metrics. RUS+SMOTE yields very poor accuracy and precision results, with significant differences of 50%± 5%, but recall gets 68% and F-measure gives a very low value i.e., 33%. There is significantly less difference between SMOTE (26%) and 7% less difference between the other two methods. SMOTE performs very well when compared to other methods for 95% CFIS Scenario.

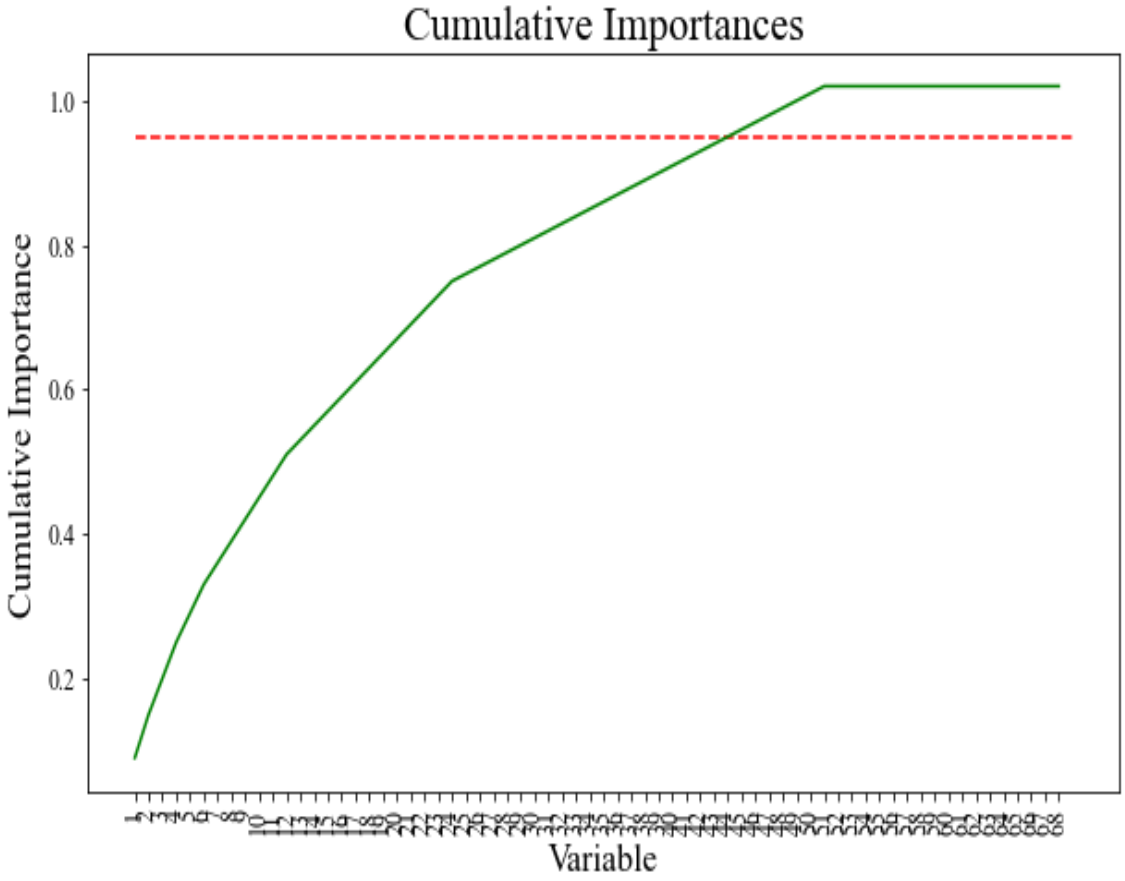


**Figure 3. Cumulative importance of the variables.**

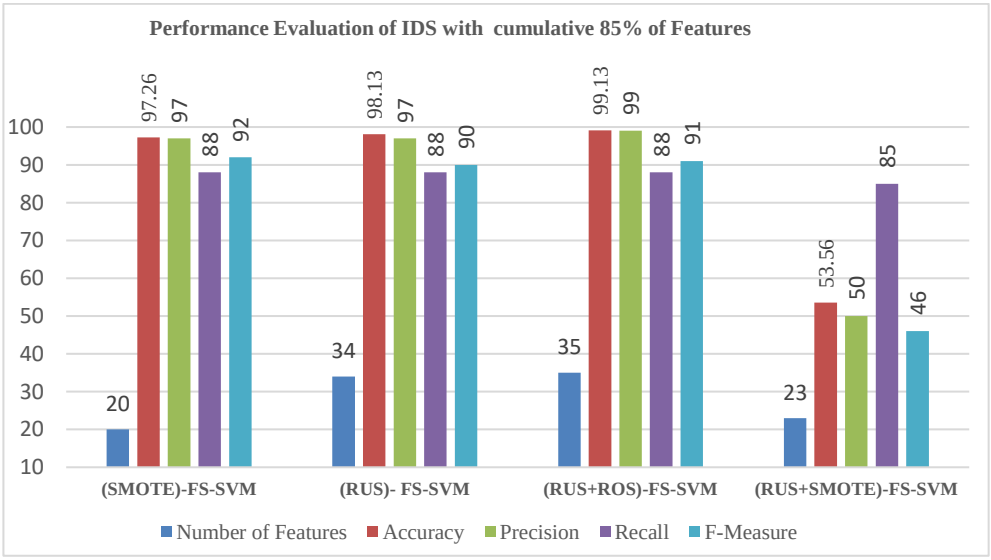


**Figure 4. Performance evaluation of IDS with 85% CFIS features.**

DOI: https://doi.org/10.52756/ijerr.2024.v43spl.004

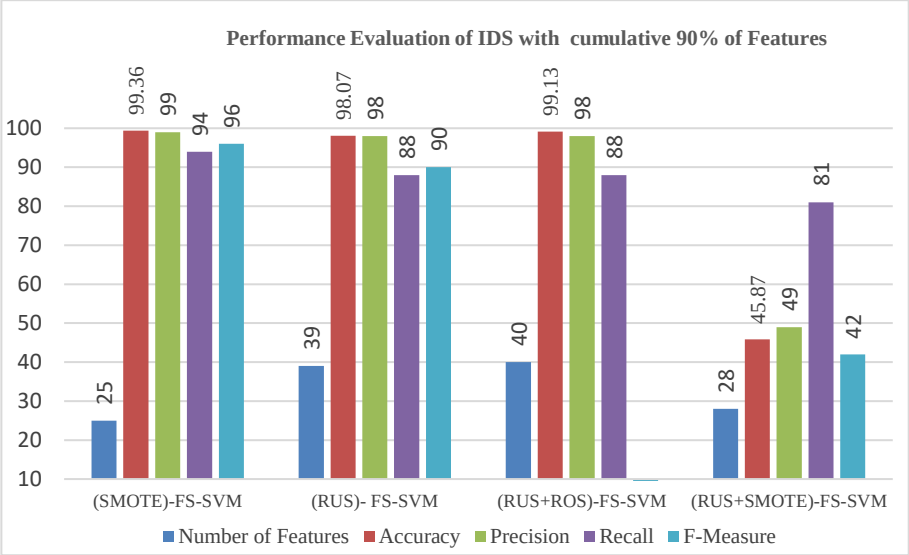


**Figure 5. Performance evaluation of IDS with 90% CFIS features.**

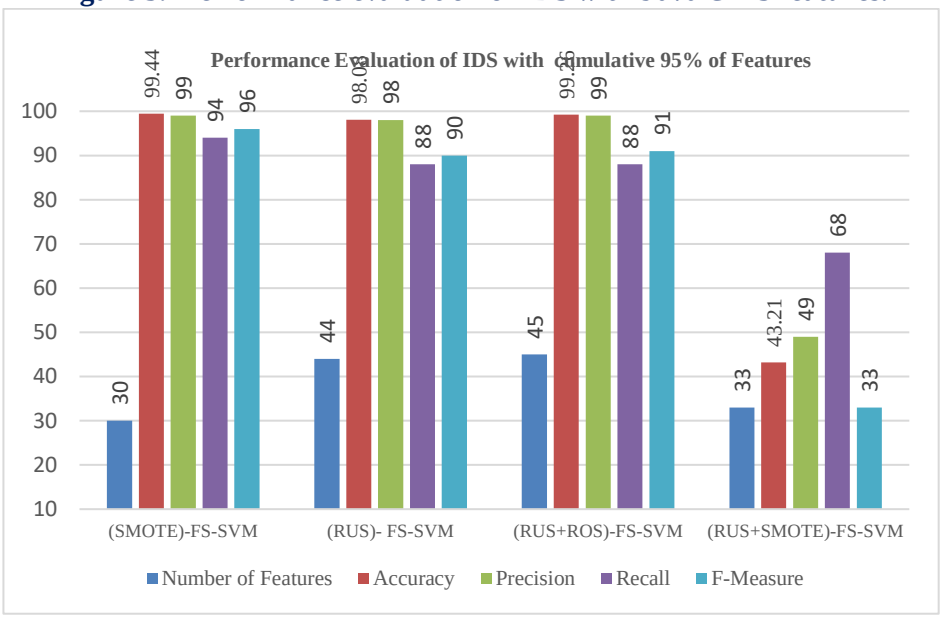


**Figure 6. Performance evaluation of IDS with 85% CFIS features.**

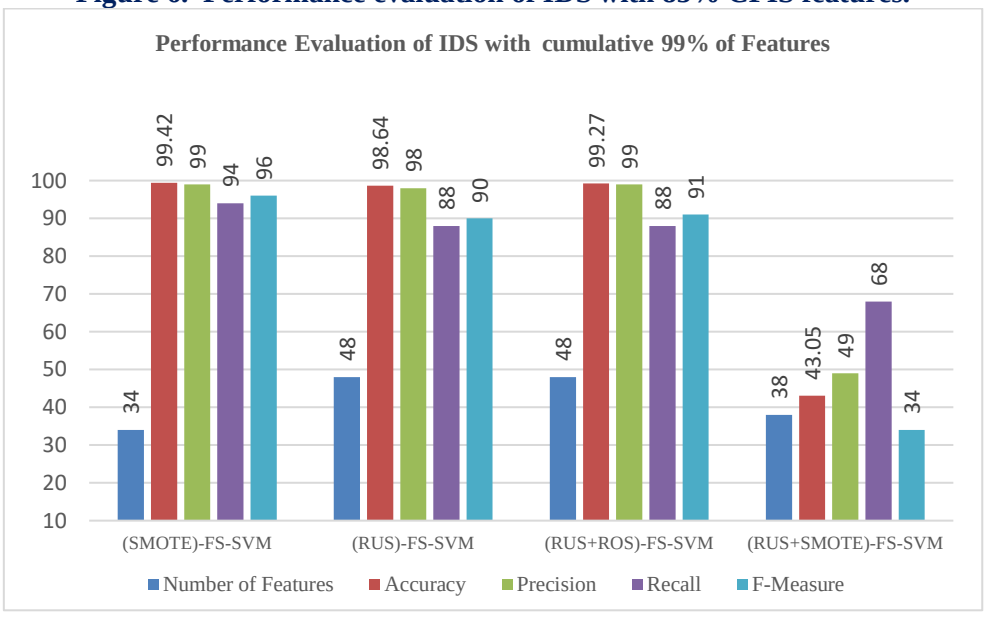


**Figure 7. Performance evaluation of IDS with 99% CFIS features.**

DOI: https://doi.org/10.52756/ijerr.2024.v43spl.004

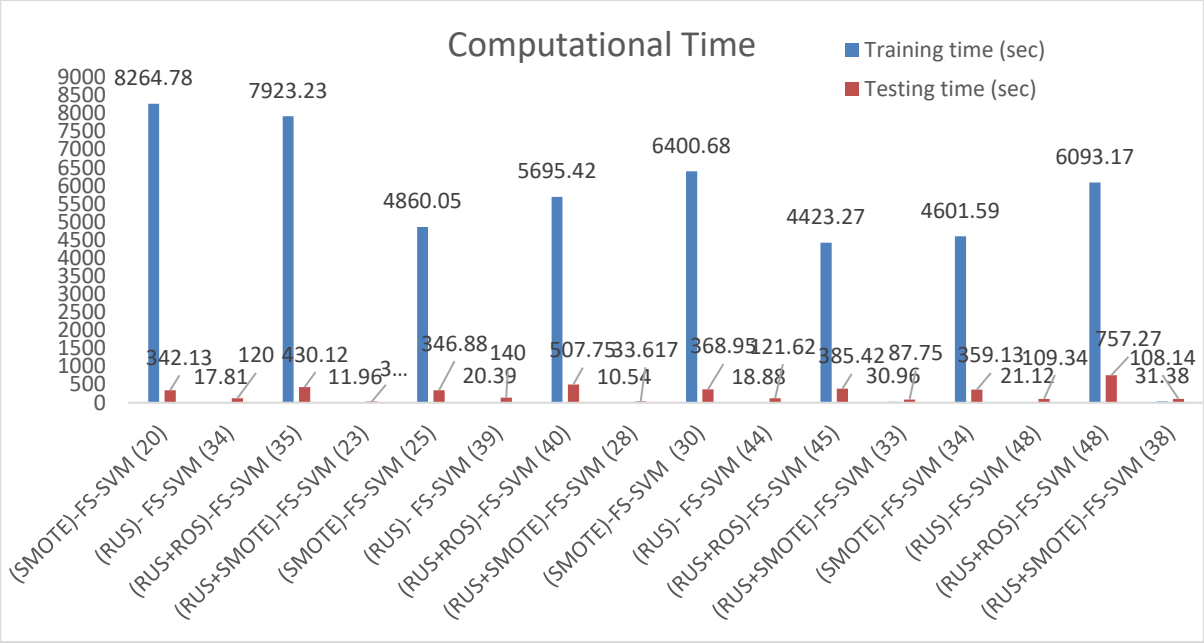


**Figure 8. Comparative analysis of computational complexity.**

### 99% CFIS Scenario

Performance analysis carried out of this scenario based on the metrics presented in the above table and their effects are shown in the above Table 6 and Figure 7. In this scenario, the SMOTE is given the same type of results of 95% CFIS and also follows the same pattern for all metrics. The other three methods show the same type of results and also follow the same pattern of order RUS+ROS, RUS and RUS+SMOTE, this will be true for all the metrics. Among the three RUS+SMOTE performs very poorly in prediction, precision and f-measure, with significant differences of 50%± 5%. However, recall gets 68% for RUS+SMOTE and 88% for the other two methods. There are significantly fewer differences from SMOTE, 20%, and 6% fewer differences from the other two methods. SMOTE provides better results when compared to other methods.

### Computational Time Analysis

In this experimental environment, the computational times are collected for a comparative study of various resampling techniques when imposing the Random Forest feature selection approach and to find the performance of these methods through the SVM classifier to address the class imbalance problem. The computational times are presented in the following Figure 8 for analysis.

From the results presented in the tables and graphs, it can be observed that the raining time of the RUS+SMOTE is less for 85% and 90% CFIS, whereas for 95% and 99% CFIS, the method RUS has less training computational times. For all 85%, 90%, 95% and 99% CFIS methods, the RUS+SMOTE is the low testing times. When comparing the total computational times, the RUS +SMOTE is given for 85%, 90% and 95% CFIS methods, whereas the RUS method in 99% CFIS gives less computational time with a marginal difference of 9 sec. The remaining methods, SMOTE and RUS+ROS methods, follow with 90% and 99% CFIS methods, but RUS+ROS and SMOTE methods follow with for 85% and 95% CFIS methods.

### Conclusion

This study analysed the performance of various resampling techniques (SMOTE, RUS, RUS+ROS and RUS+SMOTE) for addressing the class imbalance problem of NIDS through an SVM binary classifier. In addition, feature engineering was also carried out with the adoption of the Random Forest feature selection method with Bayesian Optimizer to attain different feature subsets based on CFIS thresholds.

Accuracy, Precession, Recall, F-Measure and Computational Time are chosen as performance metrics for this comparative analysis. Based on experimental

**Table 3. Experimental results with the generation of feature subsets considering CFIS are up to 85%.**

| Model | No. of Features | Accuracy | Precision | Recall | F-Measure |
|---|---|---|---|---|---|
| (SMOTE)-FS-SVM | 20 | 97.26 | 97 | 88 | 92 |
| (RUS)-FS-SVM | 34 | 98.13 | 97 | 88 | 90 |
| (RUS+ROS)-FS-SVM | 35 | 99.13 | 99 | 88 | 91 |
| (RUS+SMOTE)-FS-SVM | 23 | 53.56 | 50 | 85 | 46 |

DOI: https://doi.org/10.52756/ijerr.2024.v43spl.004

results conducted on the CIDDS-2017 dataset, it is observed that SMOTE yields the highest metric scores when compared to other methods for all CFIS threshold feature subsets. The ideal feature subset is identified with 25 features selected with 90% CFIS criteria with a marginal difference in accuracy (i.e., 0.08).

**Table 4 . Experimental results with the generation of feature subset considering CFIS is up to 90%.**

| Model | No. of Features | Accuracy | Precision | Recall | F-Measure |
|---|---|---|---|---|---|
| (SMOTE-FS-SVM | 25 | 99.36 | 99 | 94 | 96 |
| (RUS)-FS-SVM | 39 | 98.07 | 98 | 88 | 90 |
| (RUS+ROS)-FS-SVM | 40 | 99.13 | 98 | 88 | 9 |
| (RUS+SMOTE)-FS-SVM | 28 | 45.87 | 49 | 81 | 42 |

**Table 5. Experimental results with the generation of feature subset considering CFIS is up to 95%.**

| Model | No. of Features | Accuracy | Precision | Recall | F-Measure |
|---|---|---|---|---|---|
| (SMOTE)-FS-SVM | 30 | 99.44 | 99 | 94 | 96 |
| (RUS)-FS-SVM | 44 | 98.08 | 98 | 88 | 90 |
| (RUS+ROS)-FS-SVM | 45 | 99.26 | 99 | 88 | 91 |
| (RUS+SMOTE)-FS-SVM | 33 | 43.21 | 49 | 68 | 33 |

**Table 6. Experimental results with the generation of feature subset considering CFIS is up to 99%.**

| Model | No. of Features | Accuracy | Precision | Recall | F-Measure |
|---|---|---|---|---|---|
| (SMOTE-FS-SVM | 34 | 99.42 | 99 | 94 | 96 |
| (RUS)-FS-SVM | 48 | 98.64 | 98 | 88 | 90 |
| (RUS+ROS)-FS-SVM | 48 | 99.27 | 99 | 88 | 91 |
| (RUS+SMOTE)-FS-SVM | 38 | 43.05 | 49 | 68 | 34 |

This same phenomenon is followed by the RUS+ROS method and in the next order after SMOTE in terms of performance. RUS+SMOTE performance is very poor, even though the number of features is less than that of other methods. Regarding computational time, RUS+SMOTE is very minimal when compared to other methods, but the performance is also minimal, at about 50% less than that of the other methods. The RUS method is in the next place, but the accuracy performance is insignificant, with differences in various CFIS levels of other methods (1%, 1.29%, 1.36%, 0.78%). To consider the three factors in terms of performance, minimum number of features and computational time, the RUS method at 85% CFIS level with 34 features, 98.13 accuracy and minimum computational time of 137.812 sec. Finally, based on these experimental results, according to the performance view, it is concluded that the SMOTE is a better choice with a 90% CFIS level 25 feature subset. In future, one may conduct more experiments on different benchmark datasets related to different types of contemporary attacks. In addition to this, other balancing data methods on various ML and DL techniques with ROC curves were implemented to obtain experimental-based recommendation curves to suggest NIDS designers.

## Conflict of Interest

The authors declared that there is no conflict.

## References

Akgun, D., Hizal, S., & Cavusoglu, U. (2022). A new DDoS attacks intrusion detection model based on deep learning for cybersecurity. *Computers & Security*, *118*, 102748. https://doi.org/10.1016/j.cose.2022.102748

Al, S., & Dener, M. (2021). STL-HDL: A new hybrid network intrusion detection system for imbalanced dataset on big data environment. *Computers & Security*, *110*, 102435. https://doi.org/10.1016/j.cose.2021.102435

Alqarni, A. A., & El-Alfy, E. M. (2022). Improving Intrusion Detection for Imbalanced Network Traffic using Generative Deep Learning. *International Journal of Advanced Computer Science and Applications*, *13*(4), 959-967. https://doi.org/10.14569/ijacsa.2022.01304109

Awad, M., & Alabdallah, A. (2019). Addressing Imbalanced classes problem of intrusion detection system using Weighted Extreme Learning Machine. *International Journal of Computer Networks & Communications*, *11*(5), 39–58.

https://doi.org/10.5121/ijcnc.2019.11503

Babu, K. S., & Rao, Y. N. (2023). MCGAN: Modified Conditional Generative Adversarial Network (MCGAN) for class imbalance problems in Network Intrusion Detection System. *Applied Sciences*, *13*(4), 2576. https://doi.org/10.3390/app13042576

Bergstra, J., Bardenet, R., Bengio, Y., & Kégl, B. (2011). Algorithms for hyper-parameter optimization. Advances in neural information processing systems. *In Proceedings of the 24th International Conference on Neural Information Processing Systems (NIPS'11)*. Curran Associates Inc., Red Hook, NY, USA, pp. 2546–2554.

Chen, R., Dewi, C., Huang, S., & Caraka, R. E. (2020). Selecting critical features for data classification based on machine learning methods. *Journal of Big Data*, *7*, 52. https://doi.org/10.1186/s40537-020-00327-4

Chui, K. T., Gupta, B. B., Chaurasia, P., Arya, V., Almomani, A., & Alhalabi, W. (2023). Three-stage data generation algorithm for multiclass network intrusion detection with highly imbalanced dataset. *International Journal of Intelligent Networks*, *4*, 202–210. https://doi.org/10.1016/j.ijin.2023.08.001

Cui, J., Zong, L., Xie, J., & Tang, M. (2022). A novel multi-module integrated intrusion detection system for high-dimensional imbalanced data. *Applied Intelligence*, *53*(1), 272–288. https://doi.org/10.1007/s10489-022-03361-2

Elmasry, W., Akbulut, A., & Zaim, A. H. (2021). A Design of an Integrated Cloud-based Intrusion Detection System with Third Party Cloud Service. *Open Computer Science*, *11*(1), 365–379. https://doi.org/10.1515/comp-2020-0214

Fong, S., Zhuang, Y., Tang, R., Yang, X., & Deb, S. (2013). Selecting optimal feature set in High-Dimensional Data by Swarm Search. *Journal of Applied Mathematics*, *2013*, 1–18. https://doi.org/10.1155/2013/590614

Gwiazdowicz, M., & Natkaniec, M. (2023). Feature selection and model evaluation for threat detection in smart grids. *Energies*, *16*(12), 4632. https://doi.org/10.3390/en16124632

Hagar, A. A., & Gawali, B. W. (2022). Apache Spark and Deep Learning Models for High-Performance Network Intrusion Detection using CSE-CIC-IDS2018. *Computational Intelligence and Neuroscience*, *2022*, 1–11. https://doi.org/10.1155/2022/3131153

Huhn, B. (2021). What could you lose from a DDoS attack? Retrieved August 1,2024, from Citrix Blogs - Official Citrix Blogs website: https://www.citrix.com/blogs/2021/12/09/what-could-you-lose-from-a-ddos-attack/

Kudithipudi, S., Narisetty, N., Kancherla, G. R., & Bobba, B. (2023). Evaluating the efficacy of resampling techniques in addressing class imbalance for network intrusion detection systems using support vector machines. *Ingénierie Des Systèmes D Information*, *28*(5), 1229–1236. https://doi.org/10.18280/isi.280511

Kumar, N., & Sharma, S. (2013, July). Study of intrusion detection system for DDoS attacks in cloud computing. In *proceedings of the Tenth International Conference on Wireless and Optical Communications Networks (WOCN, 2013)*, pp. 1-5. DOI: 10.1109/WOCN.2013.6616255

Madhuri, T. N. P., Rao, M. S., Santosh, P. S., Tejaswi, P., & Devendra, S. (2022). Data Communication Protocol using Elliptic Curve Cryptography for Wireless Body Area Network. *In proceedings of the 6th International Conference on Computing Methodologies and Communication (ICCMC)*, 29-31 March 2022, pp.133-139. https://doi.org/10.1109/iccmc53470.2022.9753898

Mbow, M., Koide, H., & Sakurai, K. (2022). Handling class Imbalance problem in Intrusion Detection System based on deep learning. *International Journal of Networking and Computing*, *12*(2), 467–492. https://doi.org/10.15803/ijnc.12.2_467

Mijalkovic, J., & Spognardi, A. (2022). Reducing the false negative rate in deep learning based network intrusion detection systems. *Algorithms*, *15*(8), 258. https://doi.org/10.3390/a15080258

Mjahed, O., Hadaj, S. E., Guarmah, E. M. E., & Mjahed, S. (2023). New Denial of Service Attacks Detection Approach Using Hybridized Deep Neural Networks and Balanced Datasets. *Computer Systems Science and Engineering*, *47*(1), 757–775. https://doi.org/10.32604/csse.2023.039111

Mohammad, A. H. (2021). Intrusion Detection Using a New Hybrid Feature Selection Model. *Intelligent Automation & Soft Computing*, *29*(3), 65–80. https://doi.org/10.32604/iasc.2021.016140

Narisetty, N., Kancherla, G. R., Bobba, B., & K.Swathi. (2021). Investigative Study of the Effect of Various Activation Functions with Stacked Autoencoder for Dimension Reduction of NIDS using SVM. *International Journal of Advanced Computer Science and Applications*, *12*(5), 152-161. https://doi.org/10.14569/ijacsa.2021.0120519

Narisetty, N., Kancherla, G. R., Bobba, B., & Swathi, K. (2021). Hybrid Intrusion Detection Method based on constraints optimized SAE and grid search based SVM-RBF on cloud. *International Journal of Computer Networks and Applications*, *8*(6), 776. https://doi.org/10.22247/ijcna/2021/210725

Nayani, A. S. K., Sekhar, C., Rao, M. S., & Rao, K. V. (2021). Enhancing image resolution and denoising using autoencoder. In *Lecture notes on data engineering and communications technologies*, pp. 649–659. https://doi.org/10.1007/978-981-15-8335-3_50

Rao, M. S., Sekhar, C., & Bhattacharyya, D. (2021). Comparative analysis of machine learning models on loan risk analysis. In *Advances in intelligent systems and computing*, pp. 81–90. https://doi.org/10.1007/978-981-15-9516-5_7

Rish, I. (2001). An Empirical Study of the Naive Bayes Classifier. In *IJCAI 2001 Workshop on Empirical Methods in Artificial Intelligence, Seattle, 4 August 2001*. pp. 41-46.

Salo, F., Nassif, A. B., & Essex, A. (2019). Dimensionality reduction with IG-PCA and ensemble classifier for network intrusion detection. *Computer Networks*, *148*, 164–175. https://doi.org/10.1016/j.comnet.2018.11.010

Sharafaldin, I., Lashkari, A. H., & Ghorbani, A. A. (2018). Toward generating a new intrusion detection dataset and intrusion traffic characterization. *In Proceedings of the 4th International Conference on Information Systems Security and Privacy (ICISSP 2018)*, pp. 108-116. https://doi.org/10.5220/0006639801080116

Soliman, O. S., & Mahmoud, A. S. (2012). A classification system for remote sensing satellite images using support vector machine with non-linear kernel functions. In *8th International Conference on Informatics and Systems (INFOS, 2012)*, pp. BIO-181.

Sulzmann, J., Fürnkranz, J., & Hüllermeier, E. (2007). On pairwise naive Bayes classifiers. In *Lecture notes in computer science*, pp. 371–381. https://doi.org/10.1007/978-3-540-74958-5_35

Wang, C., Sun, Y., Wang, W., Liu, H., & Wang, B. (2023). Hybrid Intrusion detection system based on combination of random forest and autoencoder. *Symmetry*, *15*(3), 568. https://doi.org/10.3390/sym15030568

Yang, L., & Shami, A. (2020). On hyperparameter optimization of machine learning algorithms: Theory and practice. *Neurocomputing*, *415*, 295–316. https://doi.org/10.1016/j.neucom.2020.07.061

Zekan, M., Tomičić, I., & Schatten, M. (2022). Low-sample classification in NIDS using the EC-GAN method. *JUCS - Journal of Universal Computer Science*, *28*(12), 1330–1346. https://doi.org/10.3897/jucs.85703

Zhang, G., Wang, X., Li, R., Song, Y., He, J., & Lai, J. (2020a). Network intrusion detection based on conditional Wasserstein generative adversarial network and Cost-Sensitive stacked autoencoder. *IEEE Access*, *8*, 190431–190447. https://doi.org/10.1109/access.2020.3031892

Zhang, G., Wang, X., Li, R., Song, Y., He, J., & Lai, J. (2020b). Network intrusion detection based on conditional Wasserstein generative adversarial network and Cost-Sensitive stacked autoencoder. *IEEE Access*, *8*, 190431–190447. https://doi.org/10.1109/access.2020.3031892

Zhang, H., Zhang, B., Huang, L., Zhang, Z., & Huang, H. (2023). An efficient Two-Stage network intrusion detection system in the internet of things. *Information*, *14*(2), 77. https://doi.org/10.3390/info14020077

**How to cite this Article:**

K. Swarnalatha, Nirmalajyothi Narisetty, Gangadhara Rao Kancherla and Basaveswararao Bobba (2024). Analyzing Resampling Techniques for Addressing the Class Imbalance in NIDS using SVM with Random Forest Feature Selection. International Journal of Experimental Research and Review, 43, 42-55.

**DOI :** https://doi.org/10.52756/ijerr.2024.v43spl.004