A novel boundary adaptive oversampling approach for intrusion detection

Ritinder Kaur, Neha Gupta
School of Computer Applications, Manav Rachna International Institute of Research and Studies, Faridabad, India

Article Info

Article history:

Received Apr 17, 2024
Revised Sep 11, 2024
Accepted Sep 29, 2024

Keywords:

Decision trees
Imbalanced dataset
Imbalanced learning
Intrusion detection
Oversampling techniques

ABSTRACT

Managing unbalanced datasets is a significant challenge in intrusion detection, since uncommon assaults are often obscured by the bulk of regular network traffic. In order to mitigate the effects of class imbalance and improve intrusion detection system (IDS) performance, it is necessary to use a variety of imbalanced learning algorithms. Methods of data augmentation such as adaptive synthetic sampling (ADASYN) and synthetic minority oversampling technique (SMOTE) are useful in addressing class imbalance. This paper introduces a novel technique to data resampling where decision tree-generated decision boundaries are used to conduct ADASYN on complicated and unusual samples. When this method's efficacy was evaluated using the standard NSL-KDD dataset, the accuracy of the unusual class u2r was increased to 42% and, for r2l it was improved to 83%, respectively. The UNSW-NB 15 dataset has been used for further validation of the method, and its statistical significance has been asserted by comparing the suggested method to other oversampling techniques.

Corresponding Author:

Ritinder Kaur
School of Computer Applications, Manav Rachna International Institute of Research and Studies
Sector 43, Faridabad, Haryana 121004, India
Email: ritinderkaur.sgtbimit@gmail.com

1. INTRODUCTION

Rapid technological advancements coupled with an increasing reliance on networked computer systems have led to a startling increase in cyberthreats and attacks. Intrusion detection systems (IDS) are vital to the protection of computer networks because they can identify and stop these constantly changing threats. But it might be difficult for conventional IDS techniques to identify new and advanced threats, especially when dealing with unbalanced network data. A common and important problem in intrusion detection is data imbalance, which presents a lot of difficulties for machine learning systems. Machine learning algorithms find it challenging to correctly categorise and identify possible intrusions due to the unequal distribution of classes in intrusion detection datasets [1].

Because of this, ML algorithms could find it difficult to recognise and forecast the minority class with sufficient accuracy, which further increases the false-negative rate and reduce their overall efficacy in identifying intrusions. In real-world applications, when the minority class reflects infrequent occurrences like network assaults or unauthorised access attempts, this problem is more troubling. By raising the detection rate of minority classes, imbalanced learning in IDS has shown potential in resolving this problem [2]. Either the learning algorithm itself or the training data samples may be altered to address data imbalance at the data level. Two methods are available to data-driven techniques to achieve class distribution balancing: oversampling and undersampling. To create a balanced dataset, oversampling duplicates or synthesises instances of minority classes. Techniques like synthetic minority oversampling technique (SMOTE), adaptive synthetic sampling (ADASYN), and random oversampling may be used to accomplish this. Alternatively, to get a balanced dataset, undersampling decreases the occurrences of the majority class. Undersampling methods include tomek links and random undersampling. Even though these methods are straightforward and simple to use, if they are not used carefully, they might result in overfitting and redundancy [3].

Algorithm-based methods modify the learning algorithm to increase its sensitivity to cases of minority classes. While threshold shifting modifies the classification threshold to favour occurrences of minority classes, cost-sensitive learning allocates distinct misclassification costs to various classes. Additionally, ensemble methods may be used to enhance classification performance. This approach battles with extreme imbalance and significant computing costs, but it works with any algorithm that doesn't modify the data.

Additionally, researchers suggested hybrid strategies like SMOTEBoost, RUSBoost, and SMOTE-ENN, which blend algorithm-based and data-driven methods to maximise each one's benefits [4]–[6]. Although these hybrid algorithms involve sophisticated implementation and parameter optimisation, they increase the performance of the classifier.

Empirical studies in the literature support the use of oversampling rather than undersampling because they enhance model generalisation over unseen data and retain information seen in minority samples [7], [8]. The most widely used oversampling technique, SMOTE [9], creates synthetic instances by interpolating between a chosen instance and its neighbour after determining the k-nearest neighbours of minority class examples. Until the appropriate balance is reached, this procedure is repeated. Despite being a popular method for handling unbalanced data, SMOTE has the following drawbacks [10]:

a) Synthetics that could be artificial and inaccurately depict the actual distribution of the minority class.
b) The unrealistic assumption that there is a linear decision boundary between classes.
c) Noisy instances may be included in nearest neighbours, which causes noise to be included in synthetic samples.
d) Because there are limited samples, it may not function effectively when handling exceedingly unusual classes.

Numerous iterations of SMOTE have been proposed to overcome these problems and improve the task's performance for the imbalanced data classification throughout the years [11]. This study emphasises on SMOTE variants that utilise decision boundaries as an approach for oversampling because they allow generation of artificial samples that are compatible with the original data distribution. Decision boundary-based oversampling techniques generate synthetic samples which are accurate and illustrative of the minority class. Different classes in a dataset are categorised using decision boundaries. Various SMOTE adaptations have emerged to tackle imbalanced classification issues and enhance decision boundary precision.

While ADASYN [12] focusses on building synthetic samples in areas with fewer minority class instances to address class overlapping, Borderline-SMOTE [13] aims to enhance classification accuracy for borderline cases by targeting the minority class's decision boundary. This is further refined by safe-level SMOTE [14], which takes into account the safe-level ratio to prevent producing noisy samples. Through sample weighting and geometric analysis, MWMOTE [15], and G-SMOTE [16] enhance decision boundary management in an indirect manner. On the other hand, K-means SMOTE [17] specifically uses clustering to generate artificial samples across a range of decision boundary areas. To enhance the effectiveness of the classifier, SVM-SMOTE [18] uses support vector machines to generate artificial samples close to the SVM-defined decision boundary. More recent techniques include sophisticated methods using neural networks and algorithms inspired by nature, as well as decision boundary computation-based oversampling [19], which creates synthetic samples by examining border regions. Various neural network adaptation techniques are used, such as Siam-IDS [20], which adapts neural networks to handle sample distances during the training phase; [21] adds misclassification cost at testing phase; SASMOTE [22], which integrates attention mechanisms for high-quality sample generation. Furthermore, by improving minority class representation across several sources, federal-based [23] and meta-learning-based [24] SMOTE techniques solve class imbalance in remote databases.

Precise decision boundaries are essential for decision-based SMOTE variations to function well. Boundaries that are unclear or noisy might impair synthetic samples and lower performance. In high-dimensional data contexts, some versions, such as SMOTE-DL [25], need variable selection in order to provide an unbiased classification. An overview of SMOTE approaches is included in Table 1, with an emphasis on the main techniques and difficulties.

The performance of the primary classifier and its adeptness to capture the decision boundaries are the main factors determining how effective these changes are. Furthermore, generation of artificial samples based on decision boundaries demands extreme caution towards introduction of noise and artefacts to the dataset. High-dimensional domains further challenge the establishment of substantial decision limits. Furthermore, as the artificial samples lie close to the decision boundary, these methods were unable to adequately examine the minority class's whole space, missing important patterns or variances.

In light of these drawbacks, the goal of this study is to provide an algorithm for adaptive decision boundary-based oversampling that avoids outliers and covers the whole minority space. In order to address unbalanced categorisation, it seeks to aid in the creation of synthetic minority class samples that are more relevant and accurate. By taking into account each minority sample separately, the suggested methodology seeks to solve the issue of inadequate samples. The decision tree, which has the innate capacity to partition feature space, is suggested as the underlying classifier. Pruning decision trees prevents overfitting and causes early termination. Targeted adaptive sampling based on decision boundary closeness can manage the complicated datsets.

The remaining half of this work is structured as follows. The suggested approach is presented in section 2, and the experimental findings are discussed in depth in section 3. This paper's study is concluded in section 4, which also addresses potential future directions.

Table 1. Comparison of decision-based SMOTE variants

Sample generation principle	Smote variants	Description	Limitations
Decision boundary-based	Borderline SMOTE, SMOTEBoost, SVM-SMOTE	Generate samples near decision boundaries to balance classes.	Noisy samples if boundary is unclear
Density/distribution-based	ADASYN, MWMOTE, G-SMOTE	Use sample density and distribution to create new instances.	May amplify noise in sparse datasets
Clustering-based	KMeans-SMOTE	Clusters minority samples before generating synthetic examples.	Struggles with complex boundaries
Deep learning-based	Smote-DL	Utilizes deep learning models to capture complex decision boundaries.	Computationally intensive
Attention/hybrid methods	SASMOTE, Federal-based SMOTE	Leverages attention mechanisms or federated learning principles for synthetic sample generation.	Resource-intensive, data harmonization issues

2. RESEARCH METHOD

This work presented the proximity-adaptive synthetic minority over-sampling technique (PASMOTE) as a solution to the several shortcomings of SMOTE. This research used two widely used intrusion datasets, NSL-KDD and UNSW_NB_15, for its trials. An expansion of KDD Cup 99, NSL-KDD fixes some fundamental problems with its predecessor [26]. There is one normal class and four kinds of attacks: denial of service (DoS), probe, remote-to-user (r2l), and user-to-root (u2r). Though, there are many different types of assaults available in these attack classes, there is an imbalance in the number of instances of uncommon attack classes such as user-to-root and remote-to-user, compared to typical occurrences. The 20% training and test+ dataset files are used in the experiment. Table 2 displays the number of samples in each data set to help understand the frequency distribution in the various classes.

Table 2. Distribution of frequencies in NSL-KDD datasets

Data file used	Class distribution count				
	Normal	Dos	Probe	u2r	r2l
train 20%	13449	9234	2289	11	209
test+	9711	7458	2421	341	2754

The UNSW-NB15 dataset has been used for the proposed approach's validation and comparison with alternative oversampling techniques. This dataset combines recent synthetic assaults in networked systems with modern regular attacks [27]. This dataset has been labelled and consists of nine assault types with 49 attributes total, including the label. The training and testing sets of the dataset include 175341 and 82332 samples, respectively. The unequal distribution of classes in the two sets is seen in Figure 1, which further emphasises the experiment's significance. Since unusual attack worms make up a relatively small portion of the total distribution, they are difficult to identify, hence this research aims to enhance their detection.

Python programming was used to carry out the experiment on the well-known Google Colab IDE. Initially, both datasets were tuned for the study. There are 42 characteristics and 1 class label in NSL-KDD. Three of these features-the flag, the service, and the protocol type-are purely nominal. These attributes were converted to their numerical equivalents in order to maximise the learning process for the models. The numerical characteristics were then normalised using min-max normalisation to maintain the relative disparities between data points and guarantee that the range of values stays constant. Using formula (1), the data point X is rescaled to the range [a, b], usually [0,1].

$$Xscaled = \frac{X-Xmin}{Xmax-Xmin} \times (b-a) + a \tag{1}$$

where, Xmin represents the minimum value of the attribute, Xmax is the maximum value of the attribute, Xscaled denotes the rescaled value and the new range id defined as a – b (commonly 0 to 1).

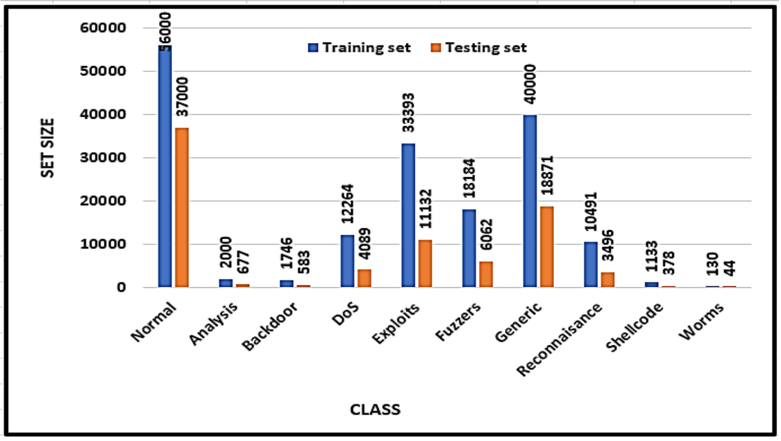
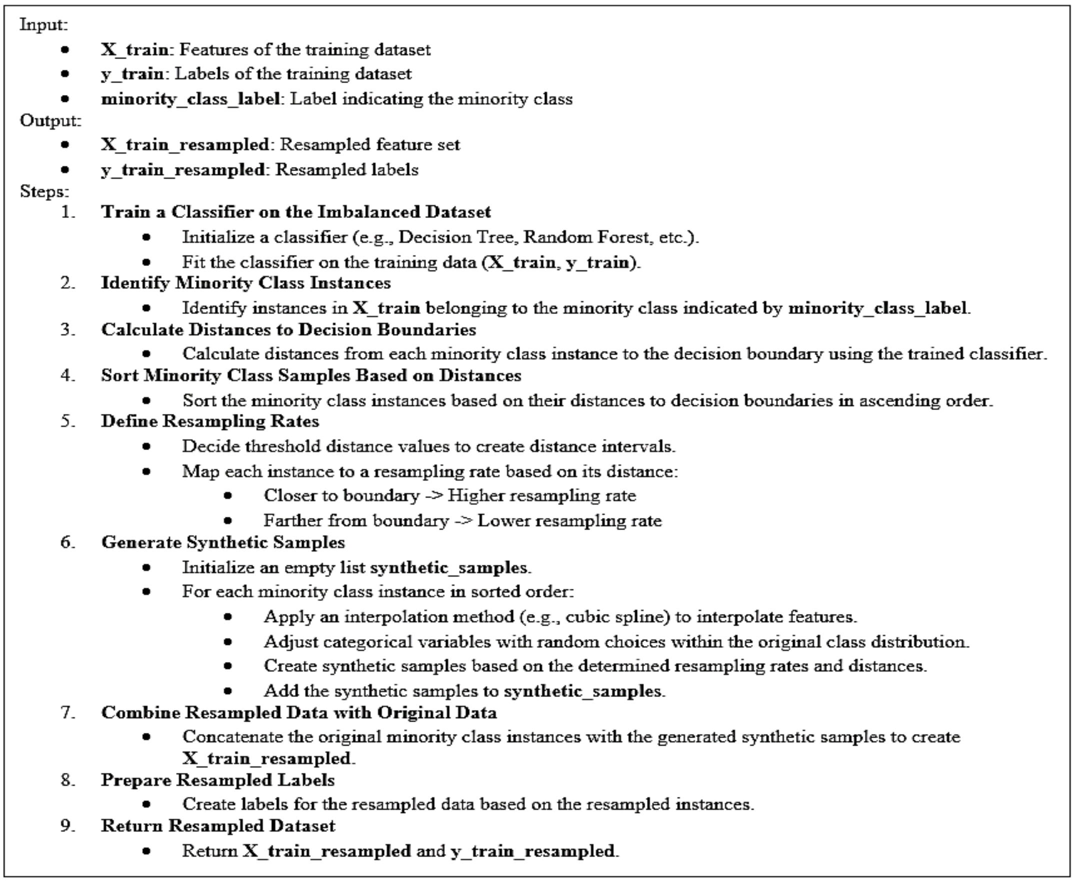


Figure 1. Distribution of samples in the UNSW-NB15 training and testing dataset

Every one of the 49 characteristics in the UNSW-NB15 dataset underwent the same kind of pretreatment. 47153 missing data led to the elimination of the service, and as the label is employed in binary investigations, it was destroyed. The categorical features attack (attack_cat), protocol (proto), and service (attack_cat) were converted using label encoders. Next, min-max scaling was used on each and every numerical characteristic. The problem was converted into a binary class problem and the rarest class worms and normal samples were extracted in order to confirm the statistical significance of the results.

To balance the NSL-KDD training dataset, the recommended PASMOTE technique Figure 2 was used. It selectively samples instances from both minority and normal classes to facilitate the training of a classifier and, thereby clarifying the decision borders and boundaries. These results are further used to produce fake samples during the resampling process.



```
Input:
    • X_train: Features of the training dataset
    • y_train: Labels of the training dataset
    • minority_class_label: Label indicating the minority class
Output:
    • X_train_resampled: Resampled feature set
    • y_train_resampled: Resampled labels
Steps:
    1. Train a Classifier on the Imbalanced Dataset
        • Initialize a classifier (e.g., Decision Tree, Random Forest, etc.).
        • Fit the classifier on the training data (X_train, y_train).
    2. Identify Minority Class Instances
        • Identify instances in X_train belonging to the minority class indicated by minority_class_label.
    3. Calculate Distances to Decision Boundaries
        • Calculate distances from each minority class instance to the decision boundary using the trained classifier.
    4. Sort Minority Class Samples Based on Distances
        • Sort the minority class instances based on their distances to decision boundaries in ascending order.
    5. Define Resampling Rates
        • Decide threshold distance values to create distance intervals.
        • Map each instance to a resampling rate based on its distance:
            • Closer to boundary -> Higher resampling rate
            • Farther from boundary -> Lower resampling rate
    6. Generate Synthetic Samples
        • Initialize an empty list synthetic_samples.
        • For each minority class instance in sorted order:
            • Apply an interpolation method (e.g., cubic spline) to interpolate features.
            • Adjust categorical variables with random choices within the original class distribution.
            • Create synthetic samples based on the determined resampling rates and distances.
            • Add the synthetic samples to synthetic_samples.
    7. Combine Resampled Data with Original Data
        • Concatenate the original minority class instances with the generated synthetic samples to create
          X_train_resampled.
    8. Prepare Resampled Labels
        • Create labels for the resampled data based on the resampled instances.
    9. Return Resampled Dataset
        • Return X_train_resampled and y_train_resampled.
```

Figure 2. Proposed PASMOTE algorithm

Because decision trees naturally split the feature domain during training, they support boundary proximity calculation for each data point. For every data point, they compute the distances to the decision borders and provide useful insights into how close examples are to the decision areas. These insights may be used to improve oversampling tactics. Furthermore, these models automatically prioritise features according to how crucial they are to the categorisation process. With this data, one may evaluate whether characteristics are important in differentiating minority class occurrences, which might guide feature engineering or selection procedures in high-dimensional datasets. Compared to linear classifiers, they are also more successful at modelling nonlinear connections between features and the target variable, enabling the modelling of more complicated decision boundaries.

For every minority class sample, the distance from the border was determined, and the samples were arranged according to these distances. To determine the resampling rates, threshold values may be computed based on the range of distances. The plan is to resample samples nearer the decision border at greater rates than those further away. Based on the distance threshold values, three resampling rates - 0.6, 0.4, and 0.2 -were used for this experiment. The cubic-spline interpolation technique was used to create the synthetic samples. A mathematical method called cubic spline interpolation may be used to estimate a function's values between known data points. To ensure smoothness and continuity, it builds a piecewise continuous curve out of many cubic polynomials. The process generates interpolated values between the known data points by iterating over each feature column and using cubic spline interpolation. Distinct values are randomly picked from the categorical variables (protocol_type, flag, and service) in minority class samples to guarantee generation of artificial samples within the original class distribution. The synthetic samples are represented by these interpolated numerical values and the modified category variables. Because cubic splines can manage sparse or irregularly spaced datasets and maintain the form and behaviour of synthetic data between existing data points, they are useful for handling nonlinear and complicated data interactions. In order to compute an interpolated value between (x_i,y_i) and (x_{i+1},y_{i+1}), the generic equation of the cubic spline may be expressed as (2).

$$S_i(x) = a_i + b_i(x - x_i) + c_i\,(x - x_i)^2 + d_i(x - x_i)^3 \tag{2}$$

which holds true for i = 1 …, n−1, where $x_i \leq x \leq x_{i+1}$ and each cubic function's coefficients a_i, b_i, c_i, and d_i must be found. The training 20% NSL-KDD dataset's sample distribution is shown in Table 3 both before and after resampling.

Table 3. 20% of classes were distributed before and after resampling in NSL-KDD training

Train 20% data file	Class distribution				
	Normal	Dos	Probe	r2l	u2r
Pre-resampling	13449	9234	2289	209	11
Post-resampling	13449	9234	2289	2603	842

To validate the proposed technique, two classes were extracted from the UNSW-NB15 training dataset: normal and worms. These classes show the range of sample distribution and the maximum and minimum samples that support the recommended procedure's effectiveness. The sample counts for the two classes were 56000 and 130, respectively, prior to distribution. Once PASMOTE was used to resample the minority class, the counts were altered to 56000 and 6318, respectively. To benchmark the proposed approach against conventional oversampling strategies, an equal number of fake samples were made from each of the four oversampling strategies mentioned in section 1: SMOTE, Random oversampler, ADASYN and, Borderline SMOTE.

Identifying the most pertinent characteristics in the data can minimise dimensionality and boost the effectiveness of machine learning algorithms. This paper makes use of a feature selection technique called as CFS-MHA which is an ensemble of cfsSubsetEval algorithms that explore feature space using meta-heuristic techniques [28]. Experimental study indicates that it is successful in prioritising the most representative features from complex intrusion detection datasets, hence reducing the computational complexity and length of trained models. After applying the algorithm to 42 features in the NSL-KDD dataset, 15 relevant features were extracted. The features that have been selected are: service, flag, wrong_fragment, hot, logged_in, is_guest_login, count, same_srv_rate, diff_serv_rate, dst_host_srv_count, dst_host_diff_srv_rate, dst_host_same_src_port_rate, dst_host_srv_diff_host_rate, dst_host_serror_rate, and dst_host_srv_serror_rate.

This work used a cost-sensitive random forest (RF) algorithmic adaptation to simulate the reduced NSL-KDD dataset. A modified technique called "cost-sensitive classification" adjusts the model's ability to generalise by adding domain-specific costs and considers the true impact of misclassification errors in order to handle imbalanced datasets [29]. Cost-sensitive classifiers primarily concentrate on differences in class distributions. Included with them is a cost matrix, also referred to as a misclassification cost matrix, that provides a clear breakdown of the costs related to various classification errors like false positives and false negatives. Unlike ordinary classifiers, which strive to reduce the aggregate classification error, cost-sensitive classifiers intend to minimise a cost function formed from the misclassification costs shown in the cost matrix.

RF was chosen as the primary classifier for both model training and evaluation. Due to its robustness, adaptability, and high projected accuracy, this well-known ensemble learning approach performs very well in a wide range of machine learning applications [30]. It constructs many decision trees using random selections of the training data, and then produces a class that is the average prediction (regression) or the mode of the classes (classification) of every single tree. It implies more unpredictability and boosts tree variation by considering just a fraction of the qualities at each split point in the decision tree.

The UNSW-NB15 model evaluation procedure used the same RF approach to guarantee consistency in the results. No changes to the cost matrix were made to the program in order to emphasise that the results were only traceable to the recommended oversampling method and not the learning algorithm.

The following metrics were taken into account to assess the model's classification performance:

a) Confusion matrix - it is a comprehensive presentation of model's predictions in a matrix form listing the number of true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN).

b) Accuracy - it calculates the ratio of correctly predicted instances to the total number of instances in the dataset to estimate the model's performance across all classes. It is calculated as (3):

$$\text{Accuracy} = \frac{(TP+TN)}{(TP+TN+FP+FN)} \tag{3}$$

c) Precision: this factor focusses on how well the model predicts good outcomes. It shows that the model can steer clear of erroneous positive predictions. It is computed as (4):

$$\text{Precision} = \frac{(TP)}{(TP+FP)} \tag{4}$$

d) Recall or sensitivity - it focusses on the model's ability to identify all positive instances, therebyreducing false negatives. It is calculated as (5):

$$\text{Recall} = \frac{(TP)}{(TP+FN)} \tag{5}$$

e) F1-score - it is an effective metrics used with imbalnced data as it calculates the harmonic mean between precision and recall. It is determined as (6):

$$\text{F1-score} = \frac{2\ x\ (Precision\ x\ Recall)}{(Precision+Recall)} \tag{6}$$

f) The receiver operating characteristics (ROC) curve, a graphical depiction of the trade-off between the true positive rate (sensitivity) and the false positive rate (specificity), is used to assess the effectiveness of binary classifiers. As a scalar number, AUC provides an overview of the model's overall performance within a range of threshold values varying from 0 to 1. Higher the AUC value, the better the model's ability to discriminate.

The proposed PASMOTE approach was statistically evaluated against four over samplers-SMOTE, ADASYN, borderline SMOTE, and random oversampler by employing three statistical tests for evaluating models. The experiment was conducted using the second dataset, UNSW-NB15. The cornerstone of these tests is the concept of the null hypothesis (H0) and the alternative hypothesis (H1). Assessing whether there is enough evidence to refute the null hypothesis is the goal of hypothesis testing, which determines the threshold for rejection depending on the degree of significance (α). Table 4 describes these tests' primary features and how to utilise them to compare binary algorithms.

Assuming the null hypothesis is true, the tests generated results in the form of a statistical value (T, U, and W) along with a p-value, that is the likelihood of observing a test metric that is as extreme as or more extreme than the one derived from the data. Since 0.05 is the threshold level of significance, p-values less than this may be interpreted as supporting evidence against the null hypothesis. The whole operation of the suggested model is shown in Figure 3.

Table 4. Specifics of the statistical tests used

Criterion	Paired T-test	Mann-whitney U test	Wilcoxon signed-rank test
Nature of test	Parametric	Non-parametric	Non-parametric
Type of data	Presumptively distributed disparities	Comparing two independent samples using a ranking system	Comparison of two matched samples based on ranks
Test statistics	T-statistic	U statistic (rank sum)	W statistic (sum of signed ranks)
Applicability	Used to compare matched sample means	Used to compare the median of two independent samples' distributions	Used to compare the two matched samples' distributions (median)
Sample size	Calls for a big sample size	Robust in relation to sample size	Robust in relation to sample size
Null hypothesis (H0)	There is no discernible change in the paired means	The distributions of the two samples are identical	There is no distinction between the paired observations
Alternative hypothesis (H1)	Notable variation between the paired means	Variations in the distribution between the two samples	Dissimilarity between the two observations

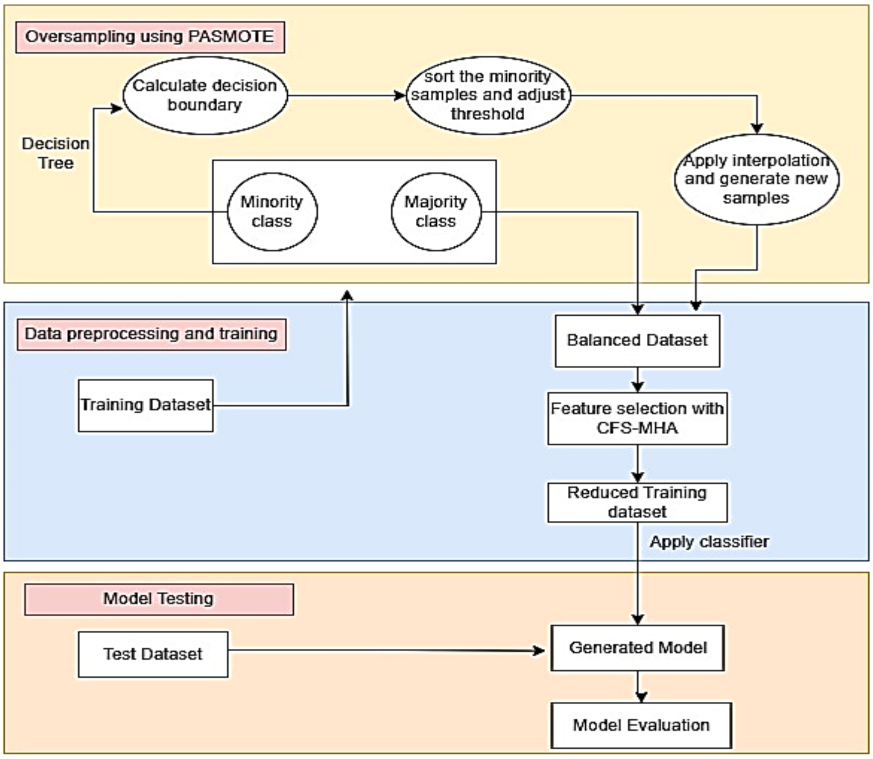

Oversampling using PASMOTE
Calculate decision boundary
sort the minority samples and adjust threshold
Decision Tree
Apply interpolation and generate new samples
Minority class
Majority class
Data preprocessing and training
Balanced Dataset
Training Dataset
Feature selection with CFS-MHA
Reduced Training dataset
Apply classifier
Model Testing
Test Dataset
Generated Model
Model Evaluation

Figure 3. Working of proposed model

3. RESULTS AND DISCUSSION

To determine the model's accuracy and generalisation potential, evaluation of the model's performance on different test data must be used. Three methods have been used in this work to investigate the model. Approach I used the first training dataset, which had unequal class distributions with all 42 characteristics, to train the model using the default hyperparameter. Using CFS-MHA, Approach II extracted 10 representative features from the unbalanced training set. Approach III used PASMOTE to oversample the training dataset, extracted 15 characteristics by feature selection, and created the assessment model. The NSL-KDD test+ dataset was then used to compare the three methods and gauge the model's performance. The effectiveness of all three of these strategies on the minority classes u2r and r2l is seen in Table 5.

Table 5. Outcomes of methods tried for minority classes in the NSL-KDD exam+

Approach	# of features		U2r samples			R2l samples		
		Overall accuracy	Accuracy	Precision	Recall	Accuracy	Precision	Recall
I	41	70.73	0.00	0.00	0.00	0.001	1.00	0.001
II	10	71.31	0.18	0.80	0.18	0.08	0.60	0.08
III	15	78.76	0.42	0.53	0.42	0.83	0.48	0.83

The results demonstrate that Approach I performed well on the dataset overall, with an overall accuracy of 70.73%, which is pretty high. Nevertheless, the performance is appalling for the minority classes (u2r and r2l). In particular, it is unable to accurately identify instances of this class, as seen by its 0% accuracy, precision, and recall when it comes to detecting u2r assaults. With respect to R2L assaults, its accuracy is somewhat higher (1.00) but its recall is very low (0.001%), indicating that it detects the bulk of R2L incidents incorrectly while accurately identifying relatively few. According to method II, the accuracy is improved to 71.31% overall by limiting the number of characteristics to 10, which is a little improvement over way I. Although there has been some progress, u2r detection levels are still rather low. Similarly, recall and accuracy of r2l detection significantly increase, but overall performance remains subpar. The technique with the best overall accuracy (78.76%) is technique III, which has 15 characteristics. This represents a substantial improvement over the earlier approaches. Compared to earlier methods, U2r detection exhibits a discernible improvement in accuracy, precision, and recall. Compared to Approach II, r2l detection exhibits a significant gain in accuracy and recall, but at the expense of somewhat lower precision. When it came to overall accuracy and advancements in identifying minority classes (U2R and R2L assaults), Approach III outperformed the other two. The results were contrasted with state-of-the-art studies in the literature to assess the effectiveness of this strategy even further. Table 6 presents the findings.

Table 6. Comparing the suggested method with the most recent research

Author	Year	Technique	Imbalanced approach	Classification strategy	Results			
					u2r		r2l	
					Precision	Recall	Precision	Recall
Douzas *et al.* [17]	2020	Siam-IDS	Adaptive neural network approach that uses Euclidian distance to calculate sample similarity	DNN, CNN	10.11	56.72	57.94	33.25
Farquad and Bose [18]	2021	GAN-based Oversampling	Produced artificial samples using Generative Adversarial Networks	Used a three-layered, cost-sensitive ANN	1	94	0	0
		KNN based oversampling	KNN interpolated between the minority samples that were already there.		2	78	23	10
Fu *et al.* [31]	2022	DLNID	ADASYN was used for oversampling	Bi-LSTM	-	24	-	65.76
Wu *et al.* [32]	2022	Enhanced RF	K-means combined with SMOTE	Improved RF using a similarity matrix	26.50	26.50	30.63	30.63
Yoon and Kim [33]	2023	SMOTE	SMOTE was used to examine the effect of feature reduction.	RF	83	7	27	28
Arık and Çavdaroğlu [34]	2023	ROGONG-IDS	To achieve balance, oversampling SMOTE and under sampling Near-Miss were combined.	XGBoost	-	10	-	39
Kaur and Gupta	2024	Proposed approach I	Absence of feature selection and oversampling technique	RF	0	0	1.0	0.001
Kaur and Gupta	2024	PASMOTE (Proposed approach III)	Using decision-boundary proximity and changing sampling threshold values to achieve oversampling	Cost-conscious RF	53	42	48	83

The comparison results demonstrate how much work is still being done in the area of unbalanced learning in intrusion detection using the NSL-KDD dataset. Even while several of the examined articles report superior overall accuracy, their biases in favour of the majority class are the main reason for their success. Douzas *et al.* [17], Farquad and Bose [18] demonstrate increased recall rates for u2r samples; nonetheless, their accuracy rates pale in comparison to the suggested methodology. On the other hand, while [33] have greater accuracy rates, their recall level is just 7%, whereas the suggested method's recall level is 42%. The literature suggests that adaptive algorithms with cost weight adjustments have also been extensively studied for managing imbalances; nevertheless, the suggested method, PASMOTE with cost-sensitive learning, performs better than the state-of-the-art techniques [18], [32].

The training dataset of UNSW-NB15 was initially oversampled using all the five oversamplers: SMOTE, ADASYN, borderline SMOTE, random oversampler, and PASMOTE. RF classifier served as the foundational classifier for both training and assessment. K-fold cross validation was carried out, with a value of k equal to 10, in order to provide accurate performance estimates and to generalise the model to encompass all cases throughout the training phase. Table 7 presents the confusion matrix values calculated using the five distinct oversampling techniques.

The performance metrics for class label worms calculated on these five models are shown in Table 8. The findings clearly reveal that PASMOTE has the greatest recall value 0.73 and precision of 0.82 leading to an elevated AUC value of 0.86 and a balanced F1-score 0.77. The random oversampler has a good recall but a poor accuracy. Although borderline SMOTE has a larger probability of false positives due to its poorer accuracy, it performs well in recall, suggesting improved detection of attack episodes. Similar performance is shown by ADASYN and SMOTE, indicating a comparable trade-off between recall and accuracy.

The ROC curve for these models is shown in Figure 4 so that the true positive and false positive rates over a range of threshold values can be seen. AUC values that are closer to 0.5 indicate random chances, while those that are closer to 1,0 indicate perfect discrimination. As a result, models nearer the top-left corner of the graph outperform other models. The notion that PASMOTE outperforms other current state-of-the-art techniques is supported by the roc curve findings.

Table 7. UNSW_NB15 dataset's confusion matrix after balancing utilizing five tried-and-true methods

	SMOTE	ADASYN	Borderline SMOTE	Random oversampler	PASMOTE
Normal as normal	36968	36966	36961	36993	36993
Normal as worms	32	34	39	7	7
Worms as normal	17	17	16	18	12
Worms as worms	27	27	28	26	32

Table 8. Findings using a RF classifier on the UNSW-NB15 test set for the worm's class

Oversampling technique	Precision	Recall	F1-score	AUC
SMOTE	0.46	0.61	0.52	0.81
ADASYN	0.44	0.61	0.51	0.81
Borderline SMOTE	0.42	0.64	0.50	0.82
Random oversampler	0.79	0.59	0.68	0.80
PASMOTE	0.82	0.73	0.77	0.86

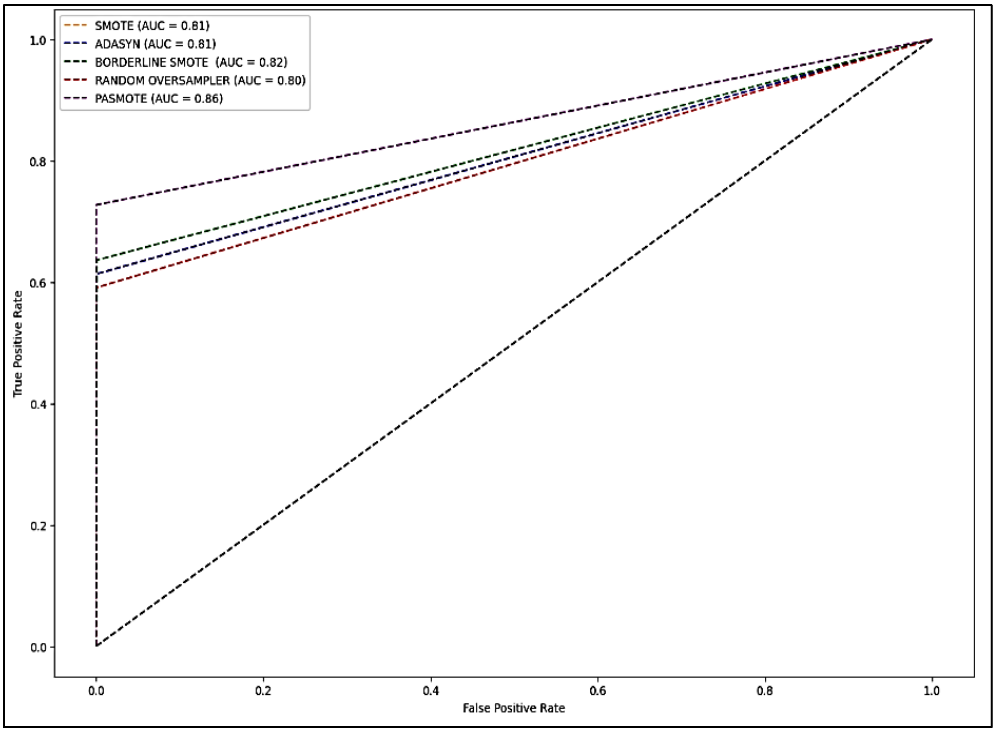


Figure 4. ROC curve for five models on UNSW-NB15 test set

These five models were subjected to statistical tests to see if the performance differences between them are statistically significant or may have happened by accident. The generalisation of these models outside of particular datasets is also aided by these tests. The three statistical tests covered in were used to evaluate the suggested model with all alternative oversampling techniques. Table 9 presents the test results along with their statistics and p-values.

Table 9. Findings from statistical tests applied on UNSW-NB15 oversampled dataset

Compared models	Paired t-test			Mann – Whitney U test			Wilcoxon test signed rank test		
	t-statistic	p-value	p < 0.05	t-statistic	p-value	p < 0.05	t-statistic	p-value	p < 0.05
PASMOTE vs. SMOTE	-3.54	0.0004	Yes	6857	0.043	Yes	99	0.0004	Yes
PASMOTE vs. ADASYN	-3.67	0.0002	Yes	6857	0.027	Yes	129	0.0002	Yes
PASMOTE vs. Borderline SMOTE	-4.54	0.0000	Yes	6856	0.006	Yes	97.5	0.0000	Yes
PASMOTE vs. Random oversampler	2.44	0.014	Yes	6862	0.479	No	0.0	0.0143	Yes

The applied statistical tests demonstrate the PASMOTE model's resilience to both skewed and normal distributions. Since all three tests and models have p-values less than 0.05, the chosen significance level for rejecting the null hypothesis, all three tests have substantially refuted the null hypothesis, which held that there is no statistical significance between these models. The lone exception is the Mann Whitney test, where less statistical significance is suggested in these models across independent samples when the p-value is greater than 0.05 between the proposed model and random over sampler.

4 CONCLUSION AND FUTURE SCOPE

The objective of this study is to increase the security of computer networks against the ever-increasing threat of cyberattacks by advancing IDS via the proposal of an adaptive oversampling technique. These versions provide a viable way to overcome unbalanced data difficulties and improve decision-making processes in several areas by including decision limits and categorisation information into the sampling process.

The suggested method showed an increase in uncommon class identification performance by using the boundary proximity idea in oversampling. This does a great job of capturing the nuances and complexity that are specific to the minority class, particularly when it comes to the decision border. The experiment investigated several threshold settings for the resampling procedure in order to optimise hyperparameters. With each cycle of oversampling, this hybrid algorithm exhibits flexibility by dynamically adjusting and choosing new instances for sampling. It is noteworthy that it places more emphasis on creating synthetic samples in areas nearer the decision border, reducing the possibility of overgeneralisation and improving the differentiation between classes. An intrinsic benefit of early halting and minimising overfitting and underfitting is provided by the use of pruned decision trees for boundary generation. Unlike SMOTE, the method does not choose sample points at random from the feature space, which lowers the possibility of adding noise to the samples that are chosen. The results of statistical testing highlight how resilient and generalisable this technique is over a wide range of areas.

Even with the algorithm's satisfactory performance, there is still room for improvement in terms of accurately detecting these unusual groups. It may be possible to improve the identification of these minority classes by fine-tuning or experimenting with other modelling strategies. It is possible to investigate the amount of realistic data generation in further research. Further research in the domain of decision-based SMOTE variants is anticipated to drive advancements in machine learning breakthroughs and result in more accurate and dependable prediction models as the area of unbalanced data sampling continues to expand. Through the use of existing tools and resources, practitioners may optimise the potential of decision-based SMOTE variants to enhance unbalanced data sampling and enhance machine learning model performance in various applications.

REFERENCES

[1] Q. Zhou, Y. Qi, H. Tang, and P. Wu, "Machine learning-based processing of unbalanced data sets for computer algorithms," *Open Computer Science*, vol. 13, no. 1, Jan. 2023, doi: 10.1515/comp-2022-0273.

[2] B. Krawczyk, "Learning from imbalanced data: open challenges and future directions," *Progress in Artificial Intelligence*, vol. 5, no. 4, pp. 221–232, Apr. 2016, doi: 10.1007/s13748-016-0094-0.

[3] D. Elreedy and A. F. Atiya, "A comprehensive analysis of synthetic minority oversampling technique (SMOTE) for handling class imbalance," *Information Sciences*, vol. 505, pp. 32–64, Dec. 2019, doi: 10.1016/j.ins.2019.07.070.

[4] N. V. Chawla, A. Lazarevic, L. O. Hall, and K. W. Bowyer, "SMOTEBoost: improving prediction of the minority class in boosting," in *Lecture Notes in Artificial Intelligence (Subseries of Lecture Notes in Computer Science)*, vol. 2838, Springer Berlin Heidelberg, 2003, pp. 107–119.

[5] C. Seiffert, T. M. Khoshgoftaar, J. Van Hulse, and A. Napolitano, "RUSBoost: a hybrid approach to alleviating class imbalance," *IEEE Transactions on Systems, Man, and Cybernetics Part A:Systems and Humans*, vol. 40, no. 1, pp. 185–197, Jan. 2010, doi: 10.1109/TSMCA.2009.2029559.

[6] M. Lamari *et al.*, "SMOTE–ENN-based data sampling and improved dynamic ensemble selection for imbalanced medical data classification," in *Advances in Intelligent Systems and Computing*, vol. 1188, Springer Singapore, 2021, pp. 37–49.

[7] R. Kaur and N. Gupta, "An empirical study on imbalanced learning in intrusion detection using random tree classifier," in *Proceedings - International Conference on Augmented Intelligence and Sustainable Systems, ICAISS 2022*, Nov. 2022, vol. 2, pp. 944–949, doi: 10.1109/ICAISS55157.2022.10010583.

[8] N. Santoso, W. Wibowo, and H. Himawati, "Integration of synthetic minority oversampling technique for imbalanced class," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 13, no. 1, pp. 102–108, Jan. 2019, doi: 10.11591/ijeecs.v13.i1.pp102-108.

[9] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, Jun. 2002, doi: 10.1613/jair.953.

[10] T. Wongvorachan, S. He, and O. Bulut, "A comparison of undersampling, oversampling, and SMOTE methods for dealing with imbalanced classification in educational data mining," *Information (Switzerland)*, vol. 14, no. 1, p. 54, Jan. 2023, doi: 10.3390/info14010054.

[11] S. Sams Aafiya Banu, B. Gopika, E. Esakki Rajan, M. P. Ramkumar, M. Mahalakshmi, and G. S. R. Emil Selvan, "SMOTE variants for data balancing in intrusion detection system using machine learning," in *Lecture Notes in Electrical Engineering*, vol. 998 LNEE, Springer Nature Singapore, 2023, pp. 317–330.

[12] Z. Chen, L. Zhou, and W. Yu, "ADASYN-random forest based intrusion detection model," in *ACM International Conference Proceeding Series*, Aug. 2021, pp. 152–159, doi: 10.1145/3483207.3483232.

[13] H. Han, W. Y. Wang, and B. H. Mao, "Borderline-SMOTE: A new over-sampling method in imbalanced data sets learning," in *Lecture Notes in Computer Science*, vol. 3644, no. PART I, Springer Berlin Heidelberg, 2005, pp. 878–887.

[14] C. Bunkhumpornpat, K. Sinapiromsaran, and C. Lursinsap, "Safe-level-SMOTE: safe-level-synthetic minority over-sampling technique for handling the class imbalanced problem," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 5476 LNAI, Springer Berlin Heidelberg, 2009, pp. 475–482.

[15] S. Barua, M. M. Islam, X. Yao, and K. Murase, "MWMOTE - majority weighted minority oversampling technique for imbalanced data set learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 2, pp. 405–425, Feb. 2014, doi: 10.1109/TKDE.2012.232.

[16] G. Douzas and F. Bacao, "Geometric SMOTE a geometrically enhanced drop-in replacement for SMOTE," *Information Sciences*, vol. 501, pp. 118–135, Oct. 2019, doi: 10.1016/j.ins.2019.06.007.

[17] G. Douzas, F. Bacao, and F. Last, "Improving imbalanced learning through a heuristic oversampling method based on k-means and SMOTE," *Information Sciences*, vol. 465, pp. 1–20, Oct. 2018, doi: 10.1016/j.ins.2018.06.056.

[18] M. A. H. Farquad and I. Bose, "Preprocessing unbalanced data using support vector machine," *Decision Support Systems*, vol. 53, no. 1, pp. 226–233, Apr. 2012, doi: 10.1016/j.dss.2012.01.016.

[19] L. Sun, M. Li, W. Ding, E. Zhang, X. Mu, and J. Xu, "AFNFS: adaptive fuzzy neighborhood-based feature selection with adaptive synthetic over-sampling for imbalanced data," *Information Sciences*, vol. 612, pp. 724–744, Oct. 2022, doi: 10.1016/j.ins.2022.08.118.

[20] P. Bedi, N. Gupta, and V. Jindal, "Siam-IDS: handling class imbalance problem in intrusion detection systems using siamese neural network," *Procedia Computer Science*, vol. 171, pp. 780–789, 2020, doi: 10.1016/j.procs.2020.04.085.

[21] S. Sapre, K. Islam, and P. Ahmadi, "A comprehensive data sampling analysis applied to the classification of rare IoT network intrusion types," in *2021 IEEE 18th Annual Consumer Communications and Networking Conference, CCNC 2021*, Jan. 2021, vol. 2, pp. 1–2, doi: 10.1109/CCNC49032.2021.9369617.

[22] A. Liu, L. Cheng, and C. Yu, "SASMOTE: a self-attention oversampling method for imbalanced CSI fingerprints in indoor positioning systems," *Sensors*, vol. 22, no. 15, p. 5677, Jul. 2022, doi: 10.3390/s22155677.

[23] Y. Liu, G. Wu, W. Zhang, and J. Li, "Federated learning-based intrusion detection on non-IID data," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 13777 LNCS, Springer Nature Switzerland, 2023, pp. 313–329.

[24] K. A. ElDahshan, A. A. A. AlHabshy, and B. I. Hameed, "Meta-heuristic optimization algorithm-based hierarchical intrusion detection system," *Computers*, vol. 11, no. 12, p. 170, Nov. 2022, doi: 10.3390/computers11120170.

[25] S. Divakar, A. Bhattacharjee, and R. Priyadarshini, "Smote-DL: a deep learning based plant disease detection method," in *2021 6th International Conference for Convergence in Technology, I2CT 2021*, Apr. 2021, pp. 1–6, doi: 10.1109/I2CT51068.2021.9417920.

[26] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," Jul. 2009, doi: 10.1109/CISDA.2009.5356528.

[27] N. Moustafa and J. Slay, "UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)," Nov. 2015, doi: 10.1109/MilCIS.2015.7348942.

[28] R. Kaur and N. Gupta, "CFS-MHA: a two-stage network intrusion detection framework," *International Journal of Information Security and Privacy*, vol. 16, no. 1, pp. 1–27, Nov. 2022, doi: 10.4018/IJISP.313663.

[29] Z. Wang, X. Chu, D. Li, H. Yang, and W. Qu, "Cost-sensitive matrixized classification learning with information entropy," *Applied Soft Computing*, vol. 116, p. 108266, Feb. 2022, doi: 10.1016/j.asoc.2021.108266.

[30] C. Maçãs, J. R. Campos, N. Lourenço, and P. Machado, "Visualisation of random forest classification," *Information Visualization*, vol. 23, no. 4, pp. 312–327, Jun. 2024, doi: 10.1177/14738716241260745.

[31] Y. Fu, Y. Du, Z. Cao, Q. Li, and W. Xiang, "A deep learning model for network intrusion detection with imbalanced data," *Electronics (Switzerland)*, vol. 11, no. 6, p. 898, Mar. 2022, doi: 10.3390/electronics11060898.

[32] T. Wu, H. Fan, H. Zhu, C. You, H. Zhou, and X. Huang, "Intrusion detection system combined enhanced random forest with SMOTE algorithm," *Eurasip Journal on Advances in Signal Processing*, vol. 2022, no. 1, May 2022, doi: 10.1186/s13634-022-00871-6.

[33] J.-E. Yoon and K. Kim, "Comparison of dimensional reduction and oversampling methods for efficient network anomaly detection," *Journal of Digital Contents Society*, vol. 24, no. 3, pp. 583–591, Mar. 2023, doi: 10.9728/dcs.2023.24.3.583.

[34] A. Arık and G. Çavdaroğlu, "An intrusion detection approach based on the combination of oversampling and undersampling algorithms," *Acta Infologica*, vol. 7, no. 1, pp. 125–138, Jun. 2023, doi: 10.26650/acin.1222890.

BIOGRAPHIES OF AUTHORS

Ritinder Kaur is pursuing her Ph.D. from Manav Rachna International Institute of Research and Studies. She is an assistant professor in Sri Guru Tegh Bahadur Institute of Management and Information Technology (GGSIPU) and has 12+ year experience in teaching. She is an active reviewer in many IEEE conferences and peer-reviewed journals. Her main focus areas in research are network attacks, intrusion systems and machine learning. She can be contacted at email: ritinderkaur.sgtbimit@gmail.com.

Dr. Neha Gupta has done her Ph.D. from Manav Rachna International University and has total of 18+ year of experience in teaching and research. She is currently working as professor at School of Computer Applications, MRIIRS. She is a life member of ACM CSTA, Tech Republic and professional member of IEEE. She has authored and coauthored 85 research papers in SCI/SCOPUS/Peer Reviewed Journals (Scopus indexed) and IEEE/IET Conference proceedings in areas of web content mining, mobile computing, and cloud computing. She has 5 national and international patents published and1 national and 2 international patents granted to her credit. She has published books with publishers like Springer, Taylor and Francis, IGI Global, and Pacific Book International and has also authored book chapters with Elsevier, Springer, CRC Press and IGI global USA. Four scholars have successfully completed their Ph.D. in her guidance. Her areas of research include cloud computing, network intrusion detection, cryptography and cyber physical systems. She is a technical programme committee (TPC) member in various conferences across globe. She can be contacted at email: neha.sca@mriu.edu.in.