Improving intrusion detection through training data augmentation.

OTOKWALA, U., PETROVSKI, A. and KALUTARAGE, H.

2021

Improving Intrusion Detection Through Training Data Augmentation

1st Uneneibotejit Otokwala
School of Computing
Robert Gordon University
Aberdeen, UK
u.otokwala@rgu.ac.uk

2nd Andrei Petrovski
School of Computing
Robert Gordon University
Aberdeen, UK
a.petrovski@rgu.ac.uk

3rd Harsha Kalutarage
School of Computing
Robert Gordon University
Aberdeen, UK
h.kalutarage@rgu.ac.uk

***Abstract*—Imbalanced classes in datasets are common problems often found in security data. Therefore, several strategies like class resampling and cost-sensitive training have been proposed to address it. In this paper, we propose a data augmentation strategy to oversample the minority classes in the dataset. Using our Sort-Augment-Combine (SAC) technique, we split the dataset into subsets of the class labels and then generate synthetic data from each of the subsets. The synthetic data were then used to oversample the minority classes. Upon the completion of the oversampling, the independent classes were combined to form an augmented training data for model fitting. Using performance metrics such as accuracy, recall (sensitivity) and true positives (specificity), the models trained using the augmented datasets show an improvement in performance metrics over the original dataset. Similarly, in a binary class dataset, SAC performed optimally and the combination of SAC and ROSE model shows an improvement in overall accuracy, sensitivity and specificity when compared with the performance of the Random Forest model on the original dataset, ROSE and SMOTE augmented datasets.**

***Index Terms*—Imbalanced data, Minority oversampling, Data augmentation, Intrusion detection**

I. Introduction

A. Background

An imbalanced dataset is a dataset in which the instance of a class or classes is (are) much more represented than the others [1]. While there is no clear-cut minimum ratio of the majority to minority classes for a dataset to be classified as imbalanced, in cybersecurity however, we have observed from our analysis of the IoT Botnet dataset [19] that a ratio of 1:3 and beyond can as well result in imbalanced dataset. As a consequence, during training, the classifier is often overwhelmed by the majority classes, resulting in an imbalanced classification [2]. S ynthetic o versampling and undersampling, cluster-based under-sampling, cost-sensitive learning, instance weighing, and other resampling techniques have been used to increase the size of the minority class or decrease the size of the majority class. However, studies by [2] have shown that the synthetic data provided by the oversampling method is often devoid of the original data's underlying structural distribution, and thus does not reduce overfitting [3]. Furthermore, when insufficient data is used during the training stages of a machine learning classifier, there are likely to be the problem of generalisation. In other words, the inability of a model to effectively generalise, i.e., to train and adjust to unseen data, usually results in bias in favour of the heavily represented majority classes [4]. This often leads to the misclassification of the classes [5]. Ineffective generalization and misclassification are costly in terms of protection because the cost of misclassifying an attack as normal or normal as attack (False Negative or False Positive) will have a significant impact on the intrusion detection system's ability to detect and prevent attacks.

Effective generalization leads to better classification, and generalization can only be improved if there is enough training data. As a result, more data is required to address the problem of generalization caused by a low data regime and imbalanced classes in a dataset. Data augmentation strategies have been implemented to increase the size of training data and the minority class(es). It has been successfully used in image classification to change the geometrical transformation of an existing image, and thus improving the image quality [6]. Non-image datasets have also been subjected to data augmentation strategies in order to increase their size. It is also worth noting that, while the strategy was found to be effective in improving generalisation and classification in some datasets, it was found to be less effective in others, owing largely to how the algorithm was applied [7]. Data augmentation in a non-image dataset, unlike image classification, will only improve generalisation and classification if the generated synthetic data has the same density and underlying distribution as the original dataset [8]. It is also worth noting that using synthetic data with a different distribution than the original distribution complicates the model's ability to effectively analyse and classify the data used [9].

Though there are a number of oversampling techniques, our choice of data augmentation strategy is with a view to enhancing effective generalisation. This is due to the fact that DDoS attacks typically result in a class imbalance between benign traffic and the attack. As a result, augmentation aids in effective generalisation, which improves classification and the model's performance in intrusion detection.

B. Motivation

We have reviewed existing work on the class imbalance problem in datasets and we are motivated:

- To use a data augmentation strategy in a cybersecurity domain for the minority class in a class distribution that is imbalanced. This strategy would necessitate the creation of synthetic datasets for augmentation, as well as a comparison of the distribution of the synthetic versus the original data.
- To see how the outcome of our data level augmented model will be especially when a tree-based algorithm like random forest is used other than a parametric algorithm. This is in view of the fact that the synthetic data may differ slightly from a majority of the original data.
- The use of data augmentation has proved very useful in image classification so we are motivated to use it to oversample the minority classes and hence improve generalization and classification, particularly in datasets with an imbalanced class distribution.

C. Contribution

To address the challenges of class imbalancing and low data regime in datasets as highlighted above, we present our contributions in this work as follows:

- A data augmentation strategy for class imbalance in datasets that can be used with both binary and multiclass datasets. This proposed novel data level data augmentation technique employs a Sort, Augment, and Combine (SAC) minority oversampling approach to address the problem of class imbalance in a dataset.
- A synthetic data that is of high perceptual quality and that has the same data distribution as the original data. This is to enhance effective blending and generalization.
- To demonstrate the SAC technique's effectiveness on the performance metrics such as sensitivity, specificity, and overall accuracy. This is due to the fact that improved generalization leads to better classification, which is critical for intrusion detection.

D. Paper organisation

The remaining sections of this paper are organised as follows: in Section 2, we highlighted related works on the subject matter. In Section 3, we put forward our methodology and the steps to achieving our contributions. In Section 4, data description and model fitting was expatiated and the results of our model was also laid out. We concluded the work in section 5 and followed up with references.

II. Related work

Because of its obvious implications, the imbalanced class problem in real-world datasets has become a huge challenge among researchers in a variety of fields, including cybersecurity [34]. As a result, numerous studies have been conducted to mitigate its impact on generalisation and classification. For example, a number of scholars have written on various resampling techniques, which could be either the undersampling of the majority class, resulting in lower training costs and the loss of vital information, or the oversampling of the minority class, which appears to increase the cost of learning. For example, Chawla et. al. [10] proposed the Synthetic Minority Over-sampling Technique (SMOTE), which can be used to generate synthetic data in order to oversample the minority class. The approach, according to the authors, could also be used to undersample the majority class in order to rebalance the classes. While this method has been lauded as a defacto approach to improving generalisation capabilities, Zhu et al. [11] opined that it does not reduce overfitting and, more importantly, that the underlying principles behind the generation of the synthetic data show that the data does not share the same distribution as the original dataset. Furthermore, Zhu et al. argued that the SMOTE technique was ineffective in dealing with a multiclass imbalance problem, resulting in over-generalization. They went on to propose a K-NN-based oversampling technique that assigns a weight relative to the nearest neighbour of the data value, such that neighbours that overgeneralize are given less weight, and thus reducing imbalance.

In their work, He et al., [12] proposed the Adaptive Synthetic Sampling (ADASYN) technique, which is used to generate synthetic data for minority classes that are difficult to learn. They believed that the technique helped to reduce biases caused by class imbalance and to shift the classification decision boundary. Similarly, Chen et al. [13] proposed the Ranked Minority Oversampling in Boosting technique in their contribution. The technique employs the idea of adaptive synthetic data generation by ranking minority class instances at each learning iteration based on the data's underlying distribution. The LRSMOTE is also a method that has been proposed as a solution to the issue of class imbalance. This technique was proposed by Liang et. al., [14], and according to the authors, the technique makes the generated synthetic data to be closed to the centre of the data sample while outliers are removed. The new data is then primed to maintain the dataset distribution. However, [15] faulted this approach as he opined that the method is designed to fit into a binary class imbalanced problem only.

Another technique proposed to address the problem of class imbalance in datasets is the cost-sensitive approach. For example, Khan et al., [15] proposed a deep neural network technique involving feature representation of both the majority and minority classes during data training. During this training, both the minority and majority classes are assigned a cost of misclassification, and the class with the higher cost is assigned a cost matrix and penalised. This strategy was also advocated by [16], but they referred to it as MetaCost. Similarly, Cao et. al., [17] proposed a cost-sensitive technique that incorporates AUC and G-mean evaluation into an objective function. Following that, SVM is used to optimise the best feature-cost parameter pairs. However, [18] was adamant about the superiority of the cost-sensitive technique over the sampling method in his argument. He also claimed that it would be difficult to test how the methods would perform

on a multiclass imbalanced dataset because the techniques are primarily designed for binary imbalanced datasets. In addition, the author hypothesised that determining the exact cost of misclassification in order to impose a penalty would be difficult in practise.

Ghazikhani et al. [21] proposed the online ensemble neural network algorithm to address class imbalance in datasets. The method employs cost-sensitive learning during the training phase, which is then followed by a weighted approach that balances the classes. In the same vein, Eke et al. [22] proposed a heterogeneous ensemble model for data resampling in an imbalanced dataset. The kernel-based mechanism was proposed [23] for improved generalisation in a binary class dataset using orthogonal forward selection (OFS) algorithms. Zhang et al. [24] in their contribution opined a flow-based intrusion detection model that combines synthetic minority oversampling and undersampling for clustering based on the Gaussian mixture Model. The use of a deep unsupervised representative learning approach was proposed by [25]. This approach, according to the authors, learns representation from data measurements and then uses an autoencoder model to translate the features to a new low-dimension representation. Vut. et al. [34] sees the problem of class imbalance as more of an overlap that clearly impacts the performance of the learning algorithm. The authors further opined that the overlap helps to deteriorate the performance at varying degrees than an imbalance does.

III. Methodology

Our approach to addressing the problem of class imbalance in datasets is centred on data augmentation through synthetic oversampling of the minority class(es). The method is based on Sort-Augment-Combine (SAC) data augmentation technique and it can be applied to both binary and multiclass datasets. The three steps involved in SAC are described below:

A. Sort

After pre-processing, the original dataset is sort into a subset of the instant classes. In other words, a binary class dataset will be sorted into two subsets of attack and benign or as the case may be.

Given a data frame, S and consisting of classes: $A, B, C, \ldots$ we can represent it as a power set, P(S)={A,B,C,...}, where $A \subseteq S, B \subseteq S, and\ C \subseteq S$. The expression implies that A, B, and C are the instant classes of the dataset, S, which can further be represented as a set as shown in equations (1) - (3).

$$A = \{a_1, a_2, a_3, ...\} \tag{1}$$

$$B = \{b_1, b_2, b_3, ...\} \tag{2}$$

$$C = \{c_1, c_2, c_3, ...\} \tag{3}$$

Where $a1, \ldots, b1, \ldots, c1, \ldots$ are elements of the subsets $A, B, and\ C$.

B. Augment

After dividing and sorting the data frame into subsets of the instant classes, a function generator - Syn(), from synthpop package in R was used to synthesize the data value from the original dataset's latent space to increase the minority classes. The generator uses sequential regression modelling to synthesize each variable one after the other in a dataset. It fits the data to the assumed distribution and obtains estimates of its parameters based on conditional distributions from which synthetic values are derived. For example, consider a dataset of variables $(Z_1, Z_2, ..., Z_n)$. Here the first variable to be synthesised is Z_1 however, because it lacks predictors before it, its synthetic values are therefore generated through random sampling with replacement from its original values. The succeeding variables distribution are then estimated and synthesised based on the conditional distributions of the preceding variables [20]. In our work, the function generator was used in conjunction with predefined parameters to generate high-quality synthesised data. For instance, each class subset was passed to the function with $m = 1$ (the number of synthetic versions of the observed data) and k (the number of cases in the synthesised data) taking different values according to the size of the synthetic data to be generated. The variables inherited by the subset from the universal set are essentially preserved during generation because other subsets share the variables (data co-location). This therefore, helps to maintain the distribution behind the original data variables. On the basis of this synthesis for example, a new set of synthetic data values are generated to form equations (4), (5) and (6) from equations (1), (2) and (3).

$$A = \{\overline{a_1}, \overline{a_2}, \overline{a_3}, ...\} \tag{4}$$

$$B = \{\overline{b_1}, \overline{b_2}, \overline{b_3}, ...\} \tag{5}$$

$$C = \{\overline{c_1}, \overline{c_2}, \overline{c_3}, ...\} \tag{6}$$

The minority classes are then supplemented independently. The process of augmentation is carried out by combining the generated synthetic data with the original subsets i.e.: equations (1) & (4); (2) & (5); and (3) & (6) are combined to form the augmented subsets of A, B, and C. As a result, the new augmented subsets are:

$$\overline{A} = \{a_1, a_2, a_3, \overline{a_1}, \overline{a_2}, \overline{a_3}, ...\} \tag{7}$$

$$\overline{B} = \{b_1, b_2, b_3, \overline{b_1}, \overline{b_2}, \overline{b_3}, ...\} \tag{8}$$

$$\overline{C} = \{c_1, c_2, c_3, \overline{c_1}, \overline{c_2}, \overline{c_3}, ...\} \tag{9}$$

C. Combine

At this stage, the new augmented subsets are combined to form a new training dataset (newTrainingdataset). In other words, combining equations (7), (8) and (9) would give us the new training dataset. $P(S) = \{\{\} + \{\overline{A}\} + \{\overline{B}\} + \{\overline{C}\}\}$. Because of co-sharing of the variables by the subsets, the combination of the augmented subsets is done through row-binding.

Algorithm: Sort-Augment-Combine (SAC)

1: **Load** dataset
2: **split** dataset into subsets of classLabels $(X_i, i+1, n)$
3 : **repeat**
4 : **for** i $\leftarrow$ 1: ncol $(X_i, i+1, n)$ *do*
5: **Load** Xi
6: **apply** synthetic function generator to generate $(\overline{X_i})$
7 : **end for**
8 : **Combine**$(X_i + \overline{X_i})$
9 : **repeat** step 3 : step 8 for classLabels (X_{i+1})
10: **until** $newClassLabels$ are formed
11 : **Group** (newTraining $\leftarrow$ $(allClassLabels)$)
12: **Return** $(newTrainingdataset)$

IV. Data Description and Model Fitting

Two datasets were used in this work and they are the BoT-IoT dataset [19] and the Smart grid dataset [32]. Both are multiclass datasets with class imbalance.

A. The BoT-IoT dataset

This dataset is the result of a laboratory simulation of IoT Botnet traffic with various types of attacks. This benchmark dataset was developed as a stop-gap measure for cybersecurity researchers and, more importantly, to enhance the understanding of modern evasive attacks. This dataset has gained popularity over the years due to its advantages over other benchmark datasets in terms of: redundant records leading to biased detection [26], several missing records as factors [27], and data unbalancing among constituent observations [28]. The size of the dataset is 82,332 observations and 42 variables consisting of 10 classes. Table I and Fig. 1 show the size and ratio of the classes relative to the largest class.

TABLE I
Original data size, ratio and distribution of instant classes.

	Class	Number of Observation	Ratio to largest class
1	Analysis	677	1:54
2	Backdoor	583	1:63
3	DoS	4089	1:9
4	Exploits	11132	1:3
5	Fuzzers	6062	1:6
6	Generic	18871	1:2
7	Normal	37000 (largest class)	1:1
8	Reconnaissance	3496	1:10
9	Shellcode	378	1:97
10	Worms	44	1:840

The class distribution as shown in Table I and Fig. 1, clearly show that normal traffic has the most observations, and its size is twice that of the nearest attack classes, i.e., Generic and Exploits. Aside from the Generic, Exploits, and Fuzzers classes, which have a ratio of 1:2, 1:3, and 1:6, respectively relative to the Normal class, the other classes which also constitute the attack type are significantly under-represented in comparison to the benign class. As a result, this is an imbalanced dataset that requires minority oversampling.

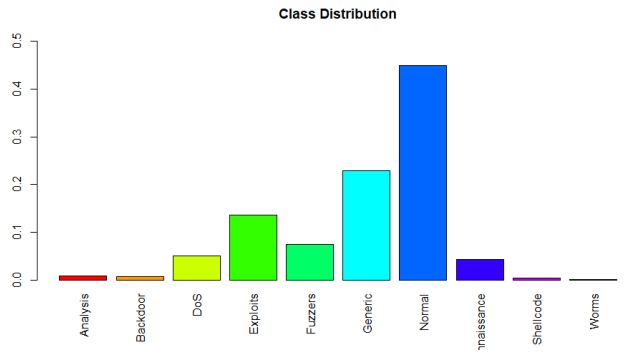

Fig. 1. showing a bar plot of class distribution in the original dataset.

We went on to transform the categorical variables using one-hot encoding, increasing the number of variables in the dataset from 42 to 187. It was now necessary to rank the variables in order of importance. To accomplish this, we used Random Forest's feature importance function, which measures the decrease in node impurity (as measured by the Gini index) as nodes are split. The ranking results show that the values of the Mean Decrease Gini range from 0.4512 to 2822. On the basis of the ranks we filtered out variables with lower importance especially variables with rankings lower than 3. Therefore, the number of variables was reduced to 53. In addition, we also used Principal Component Analysis (PCA) to obtain Principal Components (PCs) with a proportion of variance of 90%.

B. Model fitting with original dataset using Random Forest

We fitted the original dataset with a Random Forest model with K-Fold cross-validation (k=5) to enable us see and compare the confusion matrix when the oversampling was eventually done. The confusion matrix is in Fig. 2.

Confusion Matrix and Statistics

	Analysis	Backdoor	DoS	Exploits	Fuzzers	Generic	Normal	Reconnaissance	Shellcode	Worms
Analysis	42	8	8	47	39	0	0	0	0	0
Backdoor	8	0	10	27	22	2	0	3	0	0
DoS	123	45	1737	1839	249	62	20	226	3	1
Exploits	318	285	1799	7531	883	350	173	392	45	29
Fuzzers	185	213	237	765	3442	38	635	124	25	1
Generic	0	3	30	58	1	18323	13	5	6	3
Normal	1	24	209	637	1376	83	35957	363	115	5
Reconnaissance	0	4	41	200	41	4	179	2373	88	1
Shellcode	0	1	17	28	9	7	22	10	96	0
Worms	0	0	1	0	0	2	1	0	0	4

Fig. 2. Output of random forest on original dataset before minority class augmentation.

The output in Fig. 2 shows an accuracy of 84%. However, there is also a high rate of misclassification. Usually, algorithms in predictive learning always assume that classification by models is equal. The same also applies to prediction errors, where algorithms always assume that a classifier's errors are the same for all classes. This postulation only holds in an ideal situation and not in the case of an imbalanced class distribution. This is because misclassifications have a tendency to cause Type 1 and Type 2 errors [29]. Table II summarises the rate classification.

TABLE II
OVERVIEW OF MISCLASSIFICATION OF CLASSES BEFORE AUGMENTATION.

	Class	Correctly classified (%)	Misclassified (%)
1	Analysis	6	94
2	Backdoor	0	100
3	DoS	42	58
4	Exploits	67	33
5	Fuzzers	56	34
6	Generic	97	3
7	Normal	97	3
8	Reconnaissance	67	33
9	Shellcode	25	75
10	Worms	9	91

Effective classification necessitates lowering the cost of misclassification, which reduces false alarms. Apparently, effective classification cannot be achieved unless the imbalanced problem is addressed. To this end, we deployed the SAC strategy to oversample the minority classes. The size of the Generic class was used as the basis for creating the synthetic data. First, it was done to prevent model over-generalizing during training. Second, the Generic class's ratio to the Normal class being 1:2 does not indicate significant imbalance. Furthermore, the Generic class has a low misclassification rate of 3%. (see Table 2). Table III shows the new ratios after augmentation.

TABLE III
RATIO OF CLASSES TO LARGEST CLASS AFTER AUGMENTATION DATASET.

	Class	Original	Ratio	Augmented	New ratio
1	Analysis	677	1:54	18956	1:2
2	Backdoor	583	1:63	18073	1:2
3	DoS	4089	1:9	17992	1:2
4	Exploits	11132	1:3	18367	1:2
5	Fuzzers	6062	1:6	18186	1:2
6	Generic	18871	1:2	18871	1:2
7	Normal (largest)	37000	1:1	37000	1:1
8	Reconnaissance	3496	1:10	18528	1:2
9	Shellcode	378	1:97	18144	1:2
10	Worms	44	1:840	18084	1:2

The classes augmented are: Analysis, Backdoor, DoS, Exploits, Fuzzers, Reconnaissance, Shellcode, and Worms. Table III shows the new distribution.

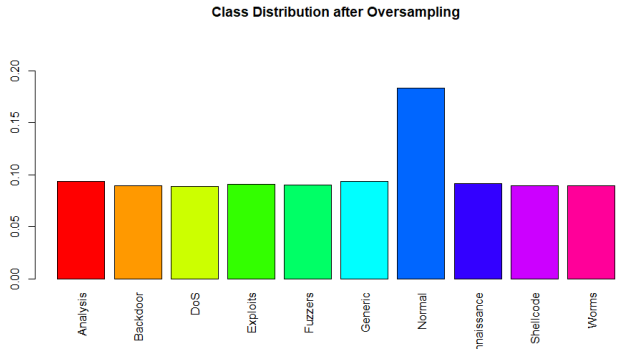

Fig. 3. Plot of the class distribution of augmented classes.

More importantly, before augmenting the minority classes with the generated data values, we ensured that the synthetic data values retained a fairly closed distribution as the original dataset's distribution. This was accomplished using a plot, as shown in Fig. 4. The figure depicts comparisons of the original and synthetic distributions of some of the classes. The dark (observed) colour represents the original data, while the light (synthetic) colour represents the generated synthetic data. The underlying structural distribution of the original class was fairly preserved in the synthetic data, as shown in the plot.

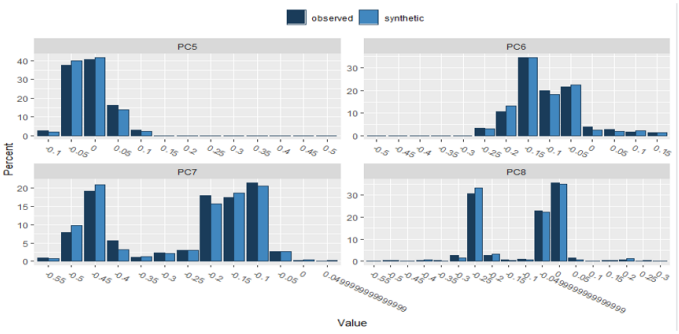

Fig. 4. A comparison between the original data and the synthetic data.

C. Model fitting with augmented minority classes dataset using Random Forest Model

Having augmented the minority classes from 82,332 to 202,200 observations with 24 features, it was then imperative to fit a model in order to observe the effect of the augmentation. Beyond that, it is instructive to also note that the challenge in engaging in malware and intrusive activities is through effective generalisation and classification. This is as Garcia et al., [30] rightly put that malware and other intrusive actors use code obfuscation methods to alter their code signatures and evade detection due to low data regime in minority classes. Data augmentation and balanced resampling are frequently used to improve generalisation and classification, particularly in techniques that use static methods for signature detection. It was thus befitting that fitting a model on a larger dataset as our augmented dataset was intended to improve generalisation and classification. To this end, we used the same fitting procedure as in the original dataset, i.e. the Random Forest model with K-Fold cross-validation, while keeping the same parameters. Fig. 5 Shows the confusion matrix.

```
               Analysis Backdoor   DoS Exploits Fuzzers Generic Normal Reconnaissance Shellcode Worms
Analysis          18066      260   350      323     655       4      4            204         0     0
Backdoor            238    17146   312      216     492       3      0             54         0     0
DoS                 148       56 14008     2459     398     105     61            441         2     1
Exploits            320      335  2274    12936     970     339    195            315         2     2
Fuzzers             182      170   358     1015   13912      50   1308            379        51     1
Generic               0        2    14       37       2   18277      9              4         5     0
Normal                1       15   182      509    1220      52  34602            272        33     3
Reconnaissance        1       27   178      563     332       9    586          16599       126     1
Shellcode             0       62   309      275     201      30    232            258     17925     1
Worms                 0        0     6       34       4       2      3              2         0 18075

Overall Statistics
```

Fig. 5. Output of random forest on oversampled minority class dataset.

The augmented minority oversampled model (Fig. 5) has an overall accuracy of 89%. When compared to the output in Fig. 2, there was an improvement in classification in terms of recall and specificity, in addition to the overall accuracy. Since

class classification has improved, it was necessary to compare the sensitivity and specificity of the original and augmented datasets. This is shown in Tables IV and V.

TABLE IV
CLASSIFICATION, MISCLASSIFICATION AND PREV. MISCLASSIFICATION.

	Class	Classifi. (%)	Misclass. (%)	Orig. Misclass. (%)
1	Analysis	95.3	4.7	94
2	Backdoor	94.8	5.7	100
3	DoS	77.8	22.2	58
4	Exploits	70.4	29.6	33
5	Fuzzers	76.4	23.6	34
6	Generic	96.8	3.2	3
7	Normal	93.5	6.5	3
8	Reconn.	89.5	10.5	33
9	Shellcode	98.7	1.3	75
10	Worms	99.9	0.01	91

Note: In Table V & VI, the row headers are:
OD - Original Data
AD - Augmented Data
DF - Difference between the sensitivity of the original value and the augmented value. The values are in percentage.

TABLE V
SHOWS THE COMPARISON BETWEEN THE SENSITIVITIES OF THE CONFUSION MATRIX OF ORIGINAL AND AUGMENTED DATASETS.

		Sensitivity									
	overall Accuracy	Analysis	Backdoor	DoS	Exploit	Fuzzers	Generic	Normal	Reconnaissance	Shellcode	Worm
OD	84	6	0	42	67	56	97	97	67	25	9
AD	89.7	95	94	77	70	76	97	93	89	98	99
DF	5.7	89	94	35	3	20	0	-4	22	73	90

Sensitivity was used to make the comparisons in Tables IV and V. The classifier's sensitivity is its ability to correctly classify the positive class (TP). From Table IV, while the misclassification for Analysis, Backdoor, Worms, Shellcode, and Dos were astronomically high in the original dataset model, it dropped to 4.7, 5.7, 0.01, 1.3 and 22.2% for the classes in the minority augmented dataset. Similarly, classification in the Fuzzers and Reconnaissance classes also improved significantly. In terms of how much improvement occurred, Table V shows the percentage difference, and with the exception of the Generic and Normal classes, which recorded a slight drop in classification, the output of the model on the minority augmented dataset shows a very significant improvement which is critical in intrusion detection.

Specificity is another important performance metric in security. It is also known as the True Negative rate (TN). It is defined as the proportion of data that are negative, and the model correctly classified them as such. A low specificity value indicates that the model classified negatives as positives incorrectly. We computed the specificity comparison from the two confusion matrices due to its obvious implications and the tendency for it to reduce the incidence of false alarm. Table VI shows the comparison.

TABLE VI
THE COMPARISON BETWEEN THE SPECIFICITY OF THE CONFUSION MATRIX OF ORIGINAL AND AUGMENTED DATASETS.

		Specificity									
	Accuracy	Analysis	Backdoor	DoS	Exploit	Fuzzers	Generic	Normal	Reconnaissance	Shellcode	Worm
OD	84	99	99	96	93	97	99	93	99	99	100
AD	89.7	99	99	98	97	98	99	98	99	99	99
DF	5.7	0	0	2	4	1	0	5	0	0	-1

Table VI shows that the augmented dataset has a slightly higher specificity than the original dataset. This is especially important in light of modern attacks that employ evasive techniques to avoid detection.

D. Using Smart Grid Dataset

This dataset is the result of a laboratory experiment. It entails measuring electrical signals on transmission lines with synchrophasors. The following parameters were measured: voltage, current, frequency, cyber-attack impedance, and normal traffic [32]. The dataset was divided into three categories: binary, triple, and multiclass, each with 15 sets of 37 event scenarios. In this work, we used two sets of the the triple class and the size of the dataset was 10,035 observations and 128 predictors. The predictors are made up of three class labels which are: Attack, Natural and NoEvents. We started by cleaning and preprocessing the data and then went further to using the Principal Component Analysis (PCA) to reduce the features from 128 to 25 Principal Components. The dataset was split into 70:30 for training and validation and the ratio of the classes relative to the largest class in the dataset is contained in Table VII.

TABLE VII
DISTRIBUTION OF THE INSTANT CLASSES IN THE ORIGINAL AND AUGMENTED SMART GRID DATASET.

	Class type	Orig. No of Observ.	Ratio	After Augmentation
1	Attack	6890	1:1	4790
2	Natural	1919	1:3	4760
3	NoEvents	495	1:13	4789

From Table VII, the relationship between the Attack class and the Natural class is not imbalanced; however, the relationship between the attack and the NoEvents class is slightly imbalanced. On this premise, we fitted a model on the original and then augmented the Natural and NoEvents classes before fitting a model on the augmented dataset. Interestingly, after the augmentation of the minority classes, the classifier was able to correctly classify 97% of the Attack class. Similarly, the classifier was also able to correctly classify the Natural and the NoEvents classes with 96% and 99% accuracy. However, there was also a 7% decrease in the classification of the

attack class after augmentation. Table VIII summarised the classification and this drop was compensated for by a 4% increase in overall accuracy. The 4% increase can be attributed to the improvement in classification in the benign classes. The significance of this is that, in intrusion detection, better classification of the benign class also contributes to fewer false alarms leading to Type 2 errors.

TABLE VIII
SHOWS THE COMPARISON BETWEEN THE SENSITIVITY AND SPECIFICITY OF THE CONFUSION MATRIX OF ORIGINAL AND AUGMENTED SMART GRID DATASETS.

		Sensitivity			Specificity		
	Overall Accuracy	Attack	Natural	NoEvents	Attack	Natural	NoEvent
Original Data	91	97	72	85	76	97	99
Augmented Data	95	90	96	99	98	95	99
Difference (%)	4	-7	24	14	22	-2	0

The improvement in the classification of the benign classes as shown in Table VIII is 24% in the Natural class and 14% in the NoEvents class. Similarly, the specificity in the attack class also improved by 22% and decreased by 2% for the benign Natural class. Notwithstanding the decrease in classification, a lower specificity is indicative of a high False Negatives which has the propensity to increase false alarms.

E. Comparing SAC, ROSE Augmented, SMOTE Augmented and SAC+ROSE Augmented datasets Using Binary dataset

Here we went a little further to see how our approach of minority oversampling method compares in binary classification. To achieve this, from the class sizes in Table VII, we formulated the Benign class by combining the Natural and NoEvents classes. The combination produced a dataset with attack and benign classes having 6890 & 2414 (74% & 26%). We then fitted a model on the dataset before and after augmentation. But first, we performed augmentation of the minority class in the binary dataset using the SAC technique, Random Over-sampling Examples (ROSE) and Synthetic Minority Oversampling Technique (SMOTE). The essence was to enable us compare how SAC compares with other minority oversampling techniques.

ROSE and SMOTE are oversampling techniques that have been used for the oversampling of the minority class in a binary dataset. Using Random Forest K-Fold (k=5) cross-validation after setting the parameters, we compared the output of the model on the original data, ROSE augmented, SMOTE augmented and the SAC augmented data. We set some of the parameters as thus:

1) TrainControl.: This parameter allows for the configuration of the number of times the cross validation will be repeated; we used "repeatedcv" to provide for consistent repeated training/testing split. We used k=5 for resampling iterations and also used random as the search tuning parameter.

2) Train.: Train. This parameter setting helps in the model fitting and it enhances the tuning process for better output. To this end, we used Random Forest (RF). As for the "tuneLength" we used 10 and also used an "ntree" of 1000.

3) Subset.: The tuning of this element enhances the auto selection of the "bestTune". This allows the selection of the best tune value from the coefficients. The bestTune value is then used as the value for "mtry".

Having configured these parameters, we augmented the dataset using ROSE package in R and also using SMOTE package in R. After the fitting of the model, we observed the comparison as shown in TABLE IX, and the output of the model with the ROSE and SMOTE augmented data were slightly better in terms of overall accuracy and sensitivity than the SAC augmented data. We then went a little further to leverage on the power of the ROSE and SAC techniques to combine their data values (SAC augmented + ROSE augmented) to form a new training data. This combination was fitted with a model and the overall accuracy, sensitivity and specificity were 98%.

TABLE IX
COMPARISON OF THE OVERALL ACCURACY, SENSITIVITY AND SPECIFICITY OF THE CONFUSION MATRIX OF ORIGINAL DATA, ROSE, SMOTE, SAC AND SAC+ROSE AUGMENTED DATASET.

Dataset	Overall Accuracy	Sensitivity	Specificity
Original data	91	98	71
ROSE Augmented	97	96	98
SMOTE Augmented	94	96	90
SAC Augmented	93	91	94
SAC + ROSE	98	98	98

From the output in TABLE IX above, we can observe that the SAC technique performed fairly high with no overfitting owing to the retention of the density and structure of the original distribution by the synthesized data values. Interestingly, the combination of the augmented SAC & ROSE datasets performed optimally with an overall accuracy, sensitivity and specificity that exceeds the other models.

F. Sensitivity and Specificity in intrusion detection

1) Sensitivity.: Sensitivity is also known as Recall or True Positive Rate (TPR). It is the proportion of actual positives of a dataset that have been predicted correctly as positive and the higher the sensitivity (recall), the better the model.

2) Specificity,: This is also known as the True Negative Rate (TNR). It the proportion of the actual negative classes of a dataset that a model is able to correctly predict as negatives.

Code:https://github.com/otokwala/Esoric/blob/master/augmentation.txt

V. CONCLUSION

The various proposals to address the problem of class imbalance in datasets were outlined in this work. We have proposed a data augmentation technique that can be used to oversample minority classes in binary and multiclass datasets

to improve generalisation and classification. Our strategy involves: Sorting, Augmenting, and then Combine (SAC). First, we divided the dataset into subsets of instant classes, and then created synthetic data from the independent subsets. We then went on to ensure that the synthetic data values have the same underlying distribution as the original dataset by using a compare function to compare the structure of the original data values and the synthetic data as a pair. Also we proceeded to supplement each independent class label in the minority class with the synthetic data. The augmented classes were then clustered to create a new training dataset. Fitting random forest model to binary and multiclass datasets significantly increased the recall and specificity. More importantly, we leveraged on the combination of SAC and ROSE to obtain an optimal classification in the binary dataset which therefore improved the overall accuracy, sensitivity, and specificity. This is significant and critical in intrusion detection and this approach can be replicated in other benchmark datasets in the future for further validation.

REFERENCES

[1] Ramyachitra, D., Manikandan, P. (2014). Imbalanced dataset classification and solutions: a review. International Journal of Computing and Business Research (IJCBR),5(4), 1-29.

[2] Sun, A., Lim, E. P., Liu, Y. (2009). On strategies for imbalanced text classification using SVM: A comparative study. Decision Support Systems, 48(1), 191-201.

[3] Meurisch, C., Bayrak, B., Giger, F., Mulhhauser M. (2020, August). PDSProxy: Trusted IoT Proxies for Confidential Ad-hoc Personalization of AI Services. In 2020 29th International Conference on Computer Communications and Networks (ICCCN) (pp. 1-2). IEEE.

[4] Lata, K., Dave, M., KN, N. (2019, February). Data augmentation using generative adversarial network. In Proceedings of 2nd International Conference on Advanced Computing and Software Engineering (ICACSE).

[5] Lemley, J., Bazrafkan, S., Corcoran, P. (2017). Smart augmentation learning an optimal data augmentation strategy. Ieee Access, 5, 5858-5869.

[6] Fadaee, M., Bisazza, A., Monz, C. (2017). Data augmentation for low-resource neural machine translation. arXiv preprint arXiv:1705.00440.

[7] Perez, L., Wang, J. (2017). The effectiveness of data augmentation in image classification using deep learning. arXiv preprint arXiv:1712.04621.

[8] Tang, B., He, H. (2015, May). Kernel ADASYN: Kernel based adaptive synthetic data generation for imbalanced learning. In 2015 IEEE Congress on Evolutionary Computation (CEC) (pp. 664-671). IEEE

[9] O'Ree, A. J., Obaidat, M. S. (2011). Security enhancements for UDDI. Security and Communication Networks, 4(8), 871-887.

[10] Chawla, N. V., Bowyer, K. W., Hall, L. O., Kegelmeyer, W. P. (2002). SMOTE:synthetic minority over-sampling technique. Journal of artificial intelligence re-search, 16, 321-357.

[11] Zhu, T., Lin, Y., Liu, Y. (2017). Synthetic minority oversampling technique formulticlass imbalance problems. Pattern Recognition, 72, 327-340.

[12] He, H., Bai, Y., Garcia, E. A., Li, S. (2008, June). ADASYN: Adaptive synthetic sampling approach for imbalanced learning. In 2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence) (pp.1322-1328). IEEE.

[13] Chen, S., He, H., Garcia, E. A. (2010). RAMOBoost: Ranked minority oversampling in boosting. IEEE Transactions on Neural Networks, 21(10), 1624-1642.

[14] Liang, X. W., Jiang, A. P., Li, T., Xue, Y. Y., Wang, G. T. (2020). LR-SMOTE—An improved unbalanced data set oversampling based on K-means and SVM. Knowledge-Based Systems, 196, 105845.

[15] Khan, S. H., Hayat, M., Bennamoun, M., Sohel, F. A., Togneri, R. (2017). Cost-sensitive learning of deep feature representations from imbalanced data. IEEE trans-actions on neural networks and learning systems, 29(8), 3573-3587.

[16] Domingos, P. (1999, August). Metacost: A general method for making classifiers cost-sensitive. In Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 155-164).

[17] Cao, P., Zhao, D., Zaiane, O. (2013, April). An optimized cost-sensitive SVM for imbalanced data learning. In Pacific-Asia conference on knowledge discovery and data mining (pp. 280-292). Springer, Berlin, Heidelberg.

[18] Weiss, G. M., McCarthy, K., Zabar, B. (2007). Cost-sensitive learning vs. sampling:Which is best for handling unbalanced classes with unequal error costs?. Dmin, 7(35-41), 24.

[19] Koroniotis, N., Moustafa, N., Sitnikova, E., Turnbull, B. (2019). Towards the development of realistic botnet dataset in the internet of things for network forensic analytics: Bot-iot dataset. Future Generation Computer Systems, 100, 779-796.

[20] Nowok, B., Raab, G. M., Snoke, J., & Dibben, C. (2016). Synthpop: generating synthetic versions of sensitive microdata for statistical disclosure control. R package version, 1-3.

[21] Ghazikhani, A., Monsefi, R., & Yazdi, H. S. (2013). Ensemble of online neural networks for non-stationary and imbalanced data streams. Neurocomputing, 122,535-544.

[22] Eke, H., Petrovski, A., & Ahriz, H. (2020). Handling minority class problem in threats detection based on heterogeneous ensemble learning approach. International Journal of Systems and Software Security and Protection (IJSSSP), 11(2), 13-37.

[23] Hong, X., Chen, S., & Harris, C. J. (2007). A kernel-based two-class classifier for imbalanced data sets. IEEE Transactions on neural networks, 18(1), 28-41.

[24] Zhang, H., Huang, L., Wu, C. Q., & Li, Z. (2020). An effective convolutional neural network based on SMOTE and Gaussian mixture model for intrusion detection in imbalanced dataset. Computer Networks, 177, 107315.

[25] Jahromi, A. N., Sakhnini, J., Karimpour, H., & Dehghantanha, A. (2019, November). A deep unsupervised representation learning approach for effective cyber-physical attack detection and identification on highly imbalanced data. In Proceedings of the 29th Annual International Conference on Computer Science and Software Engineering (pp.14-23).

[26] Mahoney, M. V., & Chan, P. K. (2003, September). An analysis of the 1999 DARPA/Lincoln Laboratory evaluation data for network anomaly detection. In International Workshop on Recent Advances in Intrusion Detection (pp. 220-237). Springer, Berlin, Heidelberg.

[27] McHugh, J. (2000). Testing intrusion detection systems: a critique of the 1998 and 1999 darpa intrusion detection system evaluations as performed by Lincoln laboratory. ACM Transactions on Information and System Security (TISSEC), 3(4),262-294.

[28] Tavallaee, M., Bagheri, E., Lu, W., & Ghorbani, A. A. (2009, July). A detailed analysis of the KDD CUP99 data set. In 2009 IEEE symposium on computational intelligence for security and defense applications (pp. 1-6). IEEE.

[29] Wankhade, K., Patka, S., & Thool, R. (2013, August). An efficient approach for intrusion detection using data mining methods. In 2013 International Conference on Advances in Computing, Communications and Informatics (ICACCI) (pp. 1615-1618). IEEE.

[30] Garcia, F. C. C., & Muga II, F. P. (2016). Random forest for malware classification. arXiv preprint arXiv:1609.07770.

[31] Boughorbel, S., Jarray, F., & El-Anbari, M. (2017). Optimal classifier for imbalanced data using Matthews Correlation Coefficient metric. PloS one, 12(6), e0177678.

[32] Pan, S., Morris, T., & Adhikari, U. (2015). Developing a hybrid intrusion detection system using data mining for power systems. IEEE Transactions on Smart Grid, 6(6), 3104-3113.

[33] Lunardon, N., Menardi, G., & Torelli, N. (2014). ROSE: A Package for Binary Imbalanced Learning. R journal, 6(1)

[34] Vuttipittayamongkol,P., Elyan, E. and Petrovski, A. (2021). On the class overlap problem in imbalanced data classification. Elsevier: Knowledge-based systems [online], 212, article number 106631. DOI: https://doi.org/10.1016/j.knosys.2020.106631.