Reducing False Negative Intrusions Rates of Ensemble Machine Learning Model based on Imbalanced Multiclass Datasets

Salim Q. Mohammed[1*] and Mohammed A. ElSheikh Hussein[2]

[1*]Department of Communication Engineering, Technical College of Engineering, Sulaimani Polytechnic University, Iraq. salim.muhammed@spu.edu.iq,
Orcid: https://orcid.org/0000-0003-3986-6701

[2]Department of Electrical Engineering, Faculty of Engineering Sciences, Sulaimani University, Iraq. mohammed.hussein@spu.edu.iq, Orcid: https://orcid.org/0000-0002-0423-7860

Received: March 03, 2023; Accepted: May 08, 2023; Published: June 30, 2023

Abstract

In spite of the efforts to improve the efficiency of intrusion detection systems based on machine learning algorithms, these systems still need more. The false negative (FN) prediction outcome is of a major priority among other outcomes, when attacks are considered as normal by classifiers. FN outputs are highly a concern issue, especially for multiclass classification, where minor classes have less instances in imbalanced datasets. In this work, three types of well-known imbalanced multiclass classification datasets are used with ensemble machine learning classifiers. The datasets: KDD99, UNSW_NB15, and CICIDS2017 are balanced using different combination of oversampling and under-sampling techniques to improve false negative rates. Suitable performance metrics have been used to obtain significant outputs improvements in all three datasets types using Random Forest classifier. Achieved accuracies are 99.9852% for KDD99, 83.5451% for UNSW_NB15 and 99.8613% for CICIDS2017. The outcomes of the work using the mentioned datasets have been compared with state-of-the-art related works and the results show a clear improvement in false negative rates.

Keywords: False Negative (FN), Intrusion Detection Systems (IDS), Supervised Machine Learning, Multiclass Classification and Random Forest Classifier (RFC).

1 Introduction

All human activities nowadays depend on communication networks(Mebawondu et al., 2020) and the security of these networks is a priority for governments, companies, as well as individuals (Ahmad et al., 2021; Daniya et al., 2021). Researchers and Academic communities had done enormous efforts to improve the security of communication networks, through using different techniques (Belouch et al., 2018; Bertoli et al., 2021). Machine learning is a profound nominee candidate to combat network attacks and mitigate intrusion on security of the networks (Almseidin et al., 2017; bhai Gupta & Agrawal, 2020; Iman & Ahmad, 2020). However, there are several challenges need to solved in order to have a secure network (Salih & Abdulazeez, 2021; Zhang et al., 2018; Zhu et al., 2017). In modern communication

Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA), volume: 14, number: 2 (June), pp. 12-30 DOI: *10.58346/JOWUA.2023.I2.002*

*Corresponding author: Department of Communication Engineering, Technical College of Engineering, Sulaimani Polytechnic University, Iraq.

networks, a huge amount of data is transmitted in different networks, small parts of these data might be a suspicious or dangerous, but the majority are normal or are licensed user's data packets (Galar et al., 2011; Krawczyk, 2016). An effective tool to separate normal and suspicious packets are machine learning based intrusion detection systems and classifiers. As most of the datasets used to train machine learning based are imbalanced, the performance of these systems has poor results, especially, for the low-class ratio records (Brownlee, 2020).

In 2015, Wahba et al. (Wahba et al., 2015) proposed an approach to improves multiclass intrusion detection systems' performance using a hybrid technique consisting of information gain and feature selection correlated by Naïve Bayes classifiers. The proposed approach has used only 13 instead of 41 features from NSL-KDD99 dataset input features. The hybrid technique is used to choose and rank the most relevant and important selected features. The work proposes a better accuracy rate and less learning time in comparison with other models.

In 2018, Keshta (Keshta, 2018) has presented a comparative study on three types of machine learning algorithms used on Intrusion Detection Systems: Support Vector Machine, Multi-layer Perceptron and Radial Basis Function. KDD99 dataset was used to evaluate the accuracy performances of these algorithms by which support Vector Machine showed higher accuracies values than other algorithms.

In 2019, Devi et al. (Ravipati & Abualkibash, 2019) have discussed several machine learning algorithms like Logistic Regression (LR), Decision Tree (DT), KNN, SVM, Random Forest (RF), Adaboost, Multilayer Perceptron, and Naïve Bayes to classify intrusion. The results showed that the RF approach has the highest performance in terms of accuracy, detection rate (DR) and false alarm rates (FAR). In the same year, (Abdulhammed et al., 2019). have proposed two techniques to reduce the input dataset features using deep learning auto-encoders and Principal Component Analysis (PCA). The CICIDS2017 dataset features have been reduced from 81 to only 10 features. The selected features are applied on different multiclass and binary classifiers like: RF, BN (Bayesian Network), QDA (Quadratic Discriminant Analysis), and LDA (Linear Discriminant Analysis) achieving intrusion detection accuracy of 99.6% for both above cases.

In 2020, (Ferrag et al., 2020) have suggested an intrusion detection system using a combined machine learning classifier for IoT (Internet of things) networks. The CICIDS2017 and BoT-IoT datasets were used to evaluate performances of several classifier models in terms of accuracy, detection rates (DR) and false alarm rates (FAR). In the same year, (Farhana et al., 2020). have devised an IDS classifier model based on deep neural network. The proposed model has been used with CICIDS2017 dataset and built by Google TensorFlow and Keras python's library. The model achieved an accuracy of 99% for multiclass and binary classifications (Liloja, 2023).

In 2020, (Fitni & Ramli, 2020). have introduced IDS based on machine learning algorithms to detect anomaly intrusion by using ensemble classifiers such as LR, DT, GB (gradient boosting) and feature selection techniques. The model performance has been improved using the CICIDS2018 dataset by selecting 23 out of 80 features achieving an accuracy of 98.8%, recall of 97.1% and F1 score of 97.9%.

Also, (Sumaiya Thaseen et al., 2021). have designed IDS model using artificial neural network and dimensionality reduction based on features correlation. Results are evaluated for multiclass classification achieving an improvement in performance metrics of accuracy, specificity and sensitivity compared with other modern techniques. In the same year, (Al-Daweri et al., 2020). have presented an analyzed KDD99 and UNSW_NB15 datasets using three different techniques RST (Rough set theory), BPNN (Back propagation neural network) and D-CFA (Discrete variant of the cuttlefish algorithm). These techniques were used to evaluate the relation between the dataset features and the output class

labels using different selection instances over multiple runs. The goal is to indicate the most effective features in these datasets.

In (Liu et al., 2021). have showed a combined IDS model based on unsupervised and supervised machine learning achieved by K-Means, Random Forest and deep neural network classifiers. Binary classification and detection have been implemented by K-Means and RF, while deep learning was used to detect attack's classes. As NSL-KDD99 and CICIDS2017 are imbalanced datasets, ADASYN (Adaptive synthetic sampling) used them. The results revealed better performances in terms of true positive ratios (TPR) for all attack's classes and less preprocessing time on training data. The proposed model obtained accuracies of 85.24% and 99.91% with NSL-KDD99 and CICIDS2017 datasets, respectively. In the same year, (Seth et al., 2021). have shown a smart IDS model based on ensemble classifiers through ranking classifier's capability in detecting attack's classes. F1 evaluation metrics was used in calculating performances of classifiers and the proposed model achieved an accuracy of 96.97% and a recall of 97.4% working with the CICIDS2018 dataset.

This paper is organization as follows. The Introduction was the first section (section 1) containing a revision on related works. The methodology is the next, followed by section 2 in which details of semi structural datasets are presented including supervised machine learning models overview block diagrams and performance metrics. Section 3 is about Performance Analysis and Results and Discussion, where the experiment results are analyzed and discussed thoroughly. Then, we have Section 4 which contains Comparisons and Conclusions sub-sections, where the outcomes of the work are compared with others state-of-the-art researches. The References section is the last one.

2 Methodology

The general block diagram of the proposed model is shown in Figure 1, which consists essentially of three types of used datasets, data pre-processing phase, separating data to training and testing sets, and classifications based on optimal performances metrics.

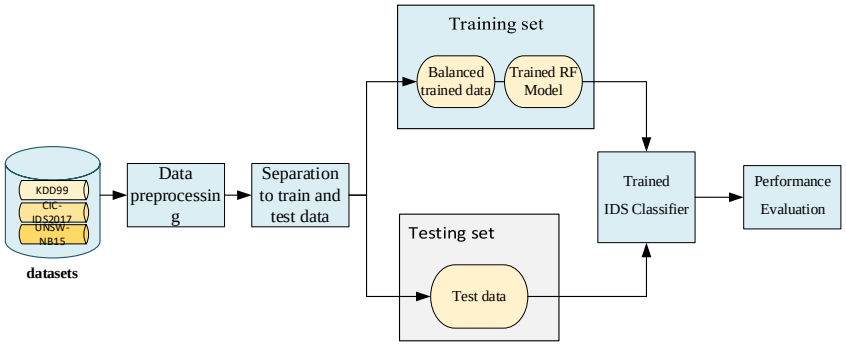


Figure 1: General Block Diagram of Proposed Model

Datasets

Three types of datasets are used with different number of records, input features, balancing of record numbers for different classes. KDD99 multiclass dataset has been used with five classes as shown in Table 1 (Hussein, 2022). This dataset is severely imbalanced among classes; the Dos attack is a major class with a ratio of 79% and the normal class is the second with a ratio of 20%. The ratio for the remaining types (Probe, R2L and U2R) class ratio is 1% altogether (Jain & Rana, 2016; Meryem & Ouahidi, 2020; Xin et al., 2018).

Table 1: KDD99 Multiclass Dataset with 41 Features

No.	Class Name	Class Type	No. of Samples	Class Ratio 100%	Training Ratio 70%	Testing Ratio 30%
1	normal	Class_0	97278	19.69	68095	29183
2	DOS	Class_1	391458	79.23	274020	117438
3	Probe	Class_2	4107	0.83	2875	1232
4	R2L	Class_3	1126	0.22	788	338
5	U2R	Class_4	52	0.01	36	16
	Total		494021		345814	148207

The second dataset is the UNSW_NB15 with 10 classes. It is different from KDD99 dataset not only in the number of classes, but also in class ratios as the normal data is the first highest major class with a ratio of 36%. In addition, the generic attack class is the second with a ratio of 22%, and as shown in Table 2 (Hooshmand & Gad, 2020; Li et al., 2018; Mishra et al., 2018).

Table 2: UNSW_NB15 Multiclass Dataset with 42 Features

No.	Class Name	Class Type	No. of Samples	Class Ratio 100%	Training Ratio 70%	Testing Ratio 30%
1	Normal	Class_0	93000	36.09	65100	27900
2	Analysis	Class_1	2677	1.03	1874	803
3	Backdoor	Class_2	2329	0.90	1630	699
4	DoS	Class_3	16353	6.34	11447	4906
5	Exploits	Class_4	44525	17.28	31167	13358
6	Fuzzers	Class_5	24246	9.40	16972	7274
7	Generic	Class_6	58871	22.84	41210	17661
8	Reconnaissance	Class_7	13987	5.42	9791	4196
9	Shellcode	Class_8	1511	0.58	1058	453
10	Worms	Class_9	174	0.06	122	52
	Total		257673		180371	77302

The CICIDS2017 multiclass dataset used in this work is completely different from the two previous datasets in having half of the records as normal ones. The highest attack class ratios for DoS Hulk, Port Scan and DDoS, are 20%, 14% and 11%, respectively (Zhou et al., 2020). Class ratios of the remaining eleven (11) attacks are less than 5% altogether, as shown in Table 3.

Table 3: CICIDS2017 Multiclass Dataset with 78 Features

No.	Class Name	Class Type	No. of Samples	Class Ratio 100%	Training Ratio 70%	Testing Ratio 30%
1	BENIGN	Class_0	557646	50	390352	167294
2	Bot	Class_1	1966	0.17	1376	590
3	DDoS	Class_2	128027	11.47	89619	38408
4	DoS Golden Eye	Class_3	10293	0.92	7205	3088
5	DoS Hulk	Class_4	231073	20.71	161751	69322
6	DoS Slow http test	Class_5	5499	0.49	3849	1650
7	DoS slow loris	Class_6	5796	0.52	4057	1739
8	FTP-Patator	Class_7	7938	0.71	5557	2381
9	Heartbleed	Class_8	11	0.00089	8	3
10	Infiltration	Class_9	36	0.0032	25	11
11	Port Scan	Class_10	158930	14.25	111251	47679
12	SSH-Patator	Class_11	5897	0.52	4128	1769
13	Web Attack: Brute Force	Class_12	1507	0.13	1055	452
14	Web Attack: Sql Injection	Class_13	21	0.0018	15	6
15	Web Attack: XSS	Class_14	652	0.058	456	196
	Total		1115292		780704	334588

Data Sampling Techniques

As mentioned before, the three types of multiclass datasets that are severely imbalanced for attack record numbers per class are used and proper identification of these classes is significantly important to ensure the security of any communication network. To improve performance of the used ensemble classifier, two factors need paying attention to: firstly, techniques used to balance the datasets; secondly the use of suitable performance metrics. Figure 2 shows the flowchart of performance metrics calculation procedure (used in this work). The first producer is the balance DS block and the dataset is balanced. Then, the decision block decides whether positive classes (also known as attack classes) are more important than the normal classes. If the answer is No, accuracy measures is calculated as a metric. If the response is Yes, the next processing step is assessing FN and FP to be followed by a decision block that examines whether FP and FN are equal to choose the proper F beta calculation measure. If the response was Yes (FN and FP were equal), the value of Beta is set to 1, and the F1-score is used as the performance metric. If FN and FP were not equal, there would be two scenarios, either calculating F0.5 or F2 score to get used as performance metrics.

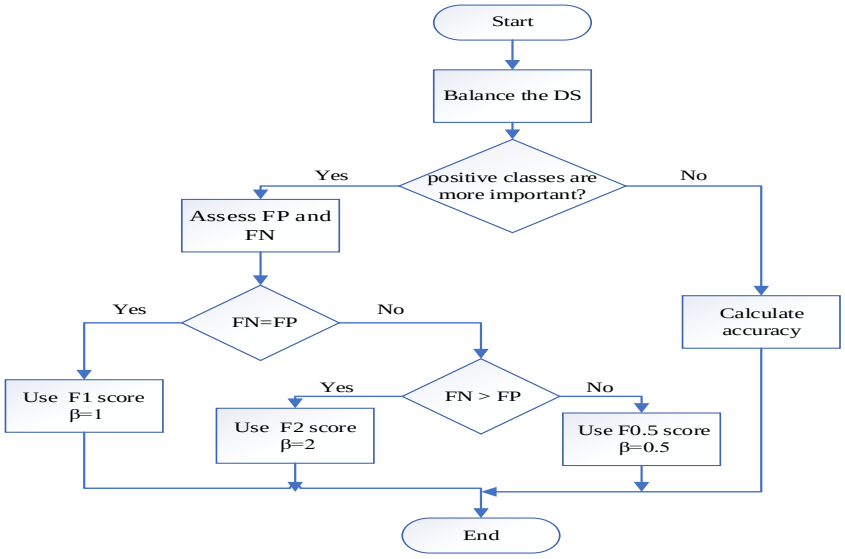


Figure 2: Flowchart of Performance Metrics Calculation

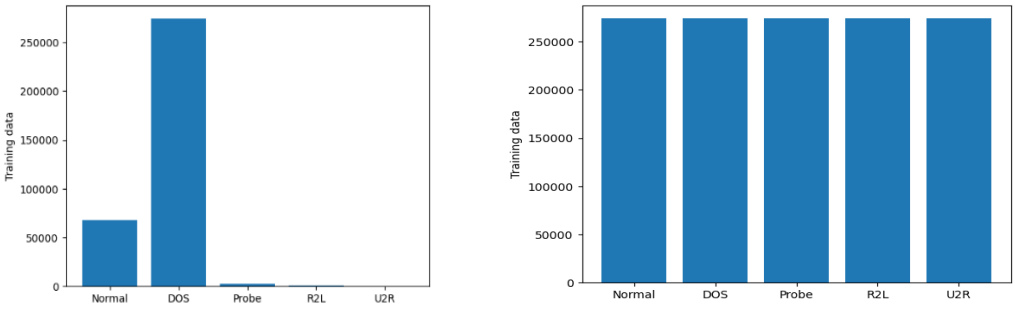
The Fβ-measure is an extension of the F-measure in which beta (β) coefficient controls the proportion of precision and recall in the computation of the evaluation metrics as shown in equation 1 (Brownlee, 2020).

$$F_\beta = \frac{(1+\beta^2) * Precision * Recall}{\beta^2 * Precision + Recall} \quad (1)$$

β value of 1 is used in this work. When false negatives are more important to reduce, Fβ-score that emphasis on recall is required and due a value of 1 has been selected for β.

To balance the KDD99 dataset, the standard combined data sampling approach SMOTE (Synthetic Minority Oversampling Technique) and Tomek Links were used. SMOTE selects instances in the training datasets that are close to one another, draws a line between them and then shows a new instance as a point along that line. The term "Tomek Links" describes a technique for locating nearest neighbors in a dataset that belong to various classes. The decision boundary in the training dataset is made less noisy or unclear by eliminating one or both of the instances in these pairings (such as the samples in the

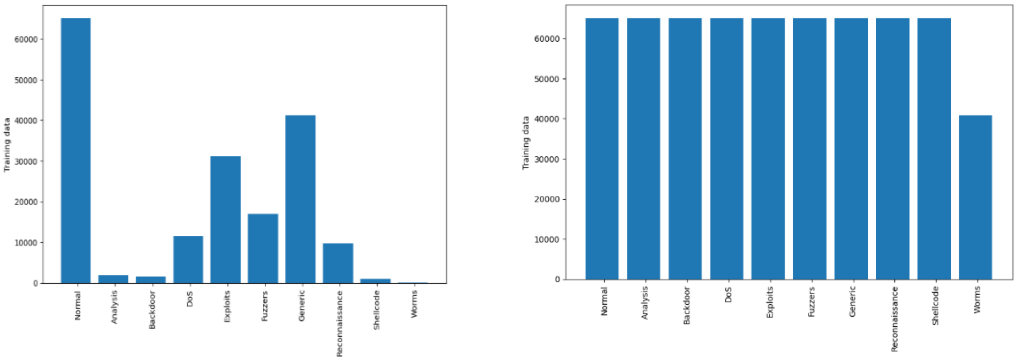
majority class). After using the SMOTE approach to oversample the minority classes (to achieve a balanced distribution), Tomek Links samples from the majority classes are found and eliminated. For a multiclass classification problem, it was found that this combination offers a reduction in false negatives at the penalty of an increase in false positives. In KDD99 dataset DOS attack is a major one. A case with 274020 records of training samples is shown in Table 1. All minority class examples are oversampled to the major class instance (DOS), except the normal class which is oversampled to 273884 records, as shown in Figure 3.



a) Imbalanced KDD99 Dataset b) Balanced KDD99 Dataset

Figure 3: Balancing of KDD99 Dataset

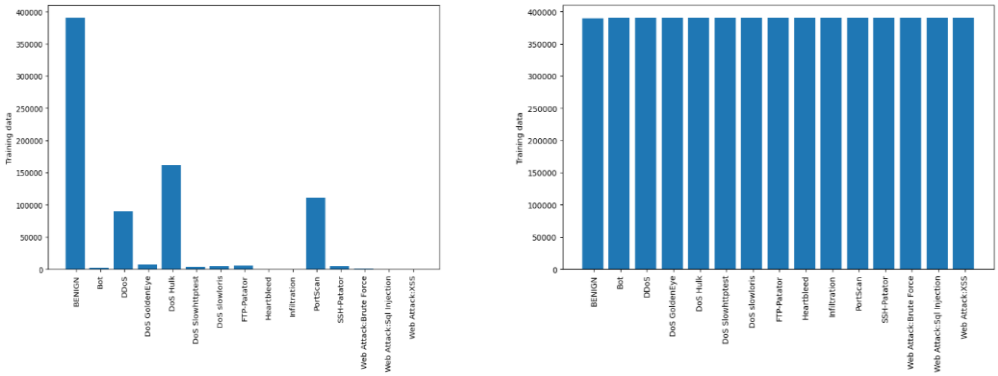
The second dataset, the UNSW_NB15 dataset has been balanced using combination of SVM and SMOTE techniques. Instances in the minority classes that are near to the support vectors are the target for synthesizing new instances when an SVM is deployed to identify the decision maximum margin indicated by the support vectors. The vectors are created after training an SVM classifier on an initial training set used to estimate the borderline area. Interpolation new instances were generated at random along the lines connecting each minority class support vector with a few of its closest neighbors. In addition to utilizing an SVM, the method aims to choose areas with fewer samples of the majority class and expands it toward the classification border. The normal class is the majority class with training data points of 65100 records. The rest eight (8) attack classes are oversampled using SMOTE approach to the same value. The least class ratio is for the Worm attack which is also oversampled to 40844 instances as shown in Figure 4.



a) Imbalanced UNSW_NB15 Dataset b) Balanced UNSW_NB15 Dataset

Figure 4: Balancing UsNSW_NB15 Dataset

In the same way, to balance the CICIDS2017 dataset, another standard of combined sampling approach is used, which combines SMOTE and ENN (Edited Nearest Neighbors). SMOTE, the most common oversampling method, can be used in conjunction with a wide range of under sampling methods like ENN which is a very well-liked under-sampling technique. Misclassified cases in a dataset are found using k = 3 nearest neighbors, and they are then eliminated. All attack classes are oversampled into the same data points of majority classes, which are 390352 instances, as in the normal class of CICIDS2017 dataset, but the normal classes are reduced to 389397 instances as shown in Figure 5.



a) Imbalanced CICIDS2017 Dataset b) Balanced CICIDS2017 Dataset

Figure 5: Balancing of the CICIDS2017 Dataset

Performance Metrics

For each class, two types of false classification predictions have been obtained. The first one is called false positive (FP), which is the normal data, but falsely predicted by the model as attack. Although, this alarm is not true, the additional investigation can be processed to ensure the data is normal. The second false detection is false negative (FN), which is the attack data. Unfortunately, the model considered it as normal data. The impact of this kind of false detection for many applications is crucial and fatal against security of the system. For the reasons mentioned previously, the best performance metrics are Precision, Recall and F1_Score, where both false positive and false negative are included in their calculations. The Precision metric is calculated from equation 2 (Sandosh et al., 2020; Subba & Gupta, 2021).

$$Precision=TP/(TP+FP) \quad (2)$$

Here, TP is true positive, which means attack data are correctly classified by the model. The false positive data points are included in Precision metric calculation for each class in the datasets. The Recall metric is evaluated from the equation 3.

$$Recall= TP/(TP+FN) \quad (3)$$

The false negative samples (FNs) are included in evaluation of the Recall metric, therefore the false negative (FN) has a high priority in reducing catastrophic impact on security of the system. The F1_Score metric is calculated from both Precision and Recall metrics as in equation 4.

$$F1\_Score=2* Precision* Recall / (Precision+ Recall) \quad (4)$$

In the same way, the overall accuracy of the model can be calculated from equation 5.

$$Accuracy=TN+TP/TN+TP+FN+FP \quad (5)$$

Here, TN is true negative data, which means that normal data are truly classified by the model.

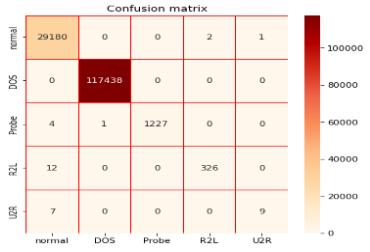
3 Performance Analysis and Results Discussion

In this section, the results and analysis of Random Forest classifier using KDD99, UNSW_NB15, and CICIDS2017 datasets are presented and the goal is multiclass classification for both balanced and imbalanced datasets. The implementation is by using Python 3.9.7 with libraries: NumPy 1.22.3, SciPy 1.7.1, Scikit-learn 1.0.2, Matplotlib 3.4.2, and Pandas 1.3.4. All runs are on virtual environment of Jupiter notebook using the configuration in shown Table 4.

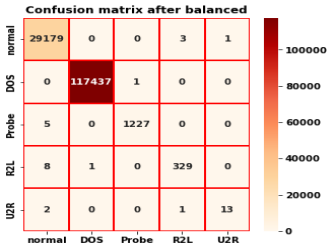
Table 4: Configuration Used in this Work

Hardware	Properties
CPU	Core i7 12th Gen 10 Core
Platform	HP VICTUS Windows 11
Ram	16GB SSD NVME
Hard	512FHD 144HZ

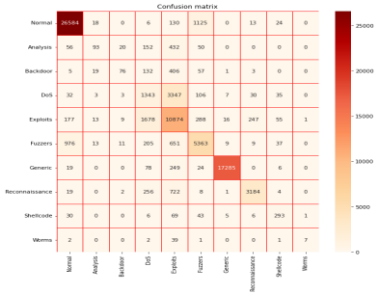
Experimental results of the datasets for both cases (balanced and imbalanced) are presented as a confusion matrix (CM) in Figure 6.



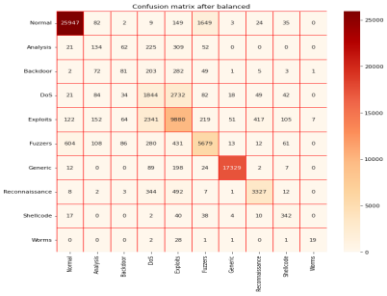
a) CM. KDD99 Imbalanced Dataset



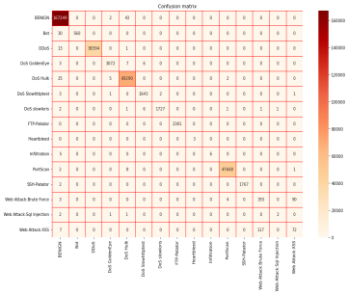
b) CM. KDD99 Balanced Dataset



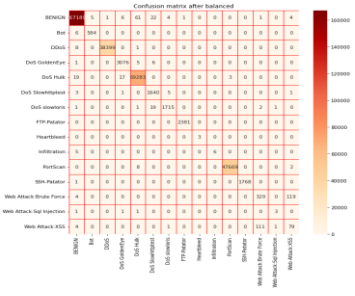
c) CM. UNSW-NB15 Imbalanced Dataset



d) CM. UNSW-NB15 Imbalanced Dataset



e) CIC-IDS2017 Imbalanced Dataset



f) CIC-IDS2017 Imbalanced Dataset

Figure 6: Confusion Matrix of Multiclass Imbalanced and Balanced Datasets

Observations on KDD99 Dataset

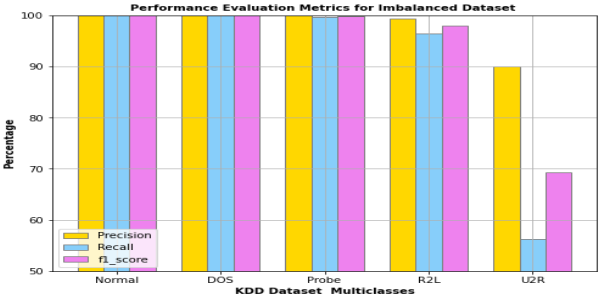
The following points are observed on KDD99 dataset performance values that are tabulated in Table 5 and shown graphically in Figure 7:

1. The model achieved an overall accuracy of 99.9852% and 99.9818% for balanced and imbalanced datasets (DS), respectively. The improvement is minor and performances metrics per class are shown in Table 5.

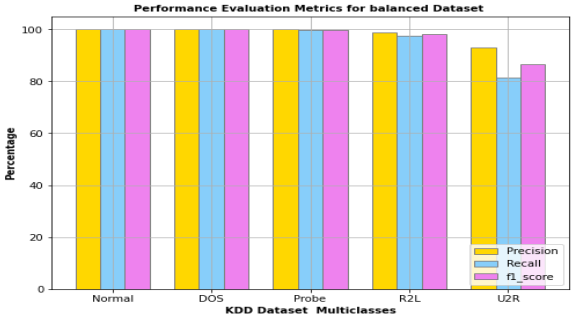
Table 5: Performances Metrics of RFC using KDD99 Balanced and Imbalanced DS

No.	Class Name	Class Ratio%	Type	Precision%	Recall%	F1_score%	FN	FP
1	DOS	79.23	imbalanced	99.9991	100	99.9996	0	0
			balanced	99.9991	99.9991	99.9991	0	1
2	normal	19.69	imbalanced	99.9897	99.9212	99.9555	0	3
			balanced	99.9863	99.9486	99.9675	0	4
3	Probe	0.83	imbalanced	100	99.5942	99.7967	4	1
			balanced	99.9186	99.5942	99.7561	5	0
4	R2L	0.22	imbalanced	99.3902	96.4497	97.8979	12	0
			balanced	98.7988	97.3373	98.0626	8	1
5	U2R	0.01	imbalanced	90	56.25	69.2308	7	0
			balanced	92.8571	81.25	86.6667	2	1
			imbalanced				23	4
			balanced				15	7

2. As expected, the first major class is the DOS attack with class ratio of 79% and it achieved the best performance, with zero false negative (FN) and false posi-tive (FP), as shown in Table 5.
3. The normal data class is with a ratio around 20% and is the second in the achieved performance. Also, it with only three false positives.
4. The R2L attack is has a ratio of 0.2% with the least achievement of twelve (12) misclassified false negatives (FN). RF graphical results is shown in Figure 7.



a) KDD99 Imbalanced DS



b) KDD99 Balanced DS

Figure 7: Performances of RFC for Balanced and Imbalanced DS

5. Having large numbers of FN is more harmful than FP and to improve classes performances a combination of SMOTE with Tomek used to duplicate samples with low class ratios and synthesize them as new samples to get augmented to low data class in the training process.
6. The DOS attack still has the highest performance value with only one false negative sample achieving a F1_Score of 99.9991%.
7. The normal data has the second level of performance value with differences for imbalanced ones as FP increased from 3 to 4 samples, achieving a F1_Score of 99.9675%.
8. The U2R attack performance value changed significantly compared to others with only two FN and one FP samples achieving a F1_Score of 86.6667%.
9. In general, performance metrics per class has improved as FN decreased from 23 to 15 and FP increased from 4 to 7.
10. Figure 8 shows graphical representation of performance metrics values.

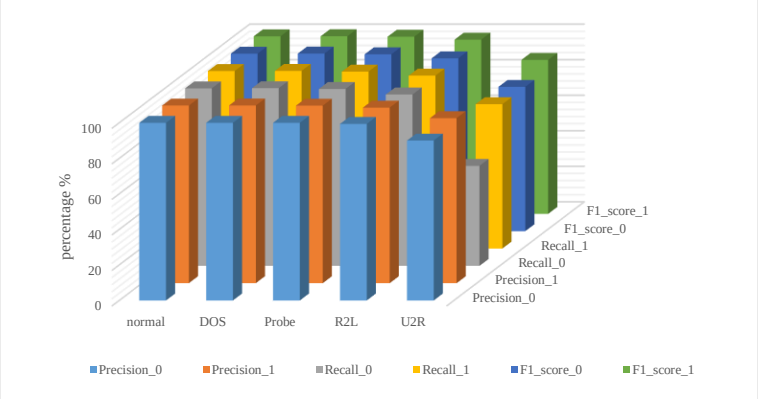


Figure 8: Performance Metrics of the KDD99 DS

Observations on UNSW_NB15 Dataset

The following points are key observations on UNSW_NB15 dataset. Performance values supported by tabular data is shown in Table 6 and graphical analyses are shown in Figure 9 and Figure 10. These outcomes are based on RFC with a combination of SMOTE plus and SVM algorithms, as follows:

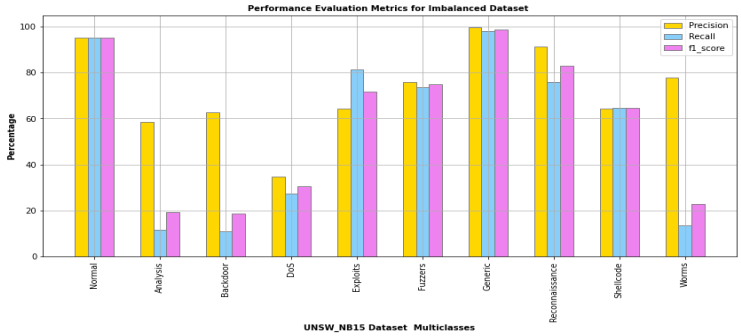
1. RFC achieved an overall accuracy of 83.5451% for balanced DS and 84.2177% for imbalanced DS with slightly decreasing overall accuracy.

Table 6: Performance Metrics using UNSW_NB15 for Imbalanced and Balanced DS

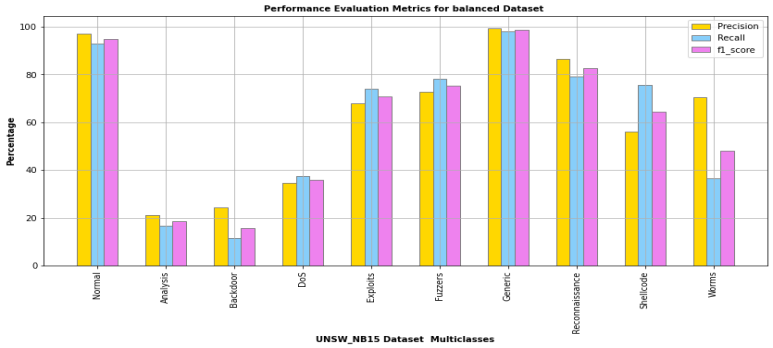
No.	Class Name	Class Ratio%	Type	Precision%	Recall%	F1_score%	FN	FP
1	Generic	22.84	imbalanced	99.7749	97.871	98.8138	19	357
			balanced	99.4719	98.1202	98.7914	12	320
2	Normal	36.09	imbalanced	95.2832	95.2832	95.2832	0	1316
			balanced	96.9836	93	94.95	0	1953
3	Reconnaissance	5.42	imbalanced	91.1798	75.8818	82.8304	19	993
			balanced	86.5055	79.2898	82.7406	8	861
4	Fuzzers	9.4	imbalanced	75.9094	73.7283	74.803	976	935
			balanced	72.8077	78.0726	75.3483	604	991
5	Exploits	17.28	imbalanced	64.2709	81.4044	71.8301	177	2307
			balanced	67.9458	73.9632	70.8269	122	3356
6	Shellcode	0.58	imbalanced	64.3956	64.6799	64.5374	30	130

			balanced	56.25	75.4967	64.4675	17	94
7	DoS	6.34	imbalanced	34.8108	27.3746	30.6481	32	3531
			balanced	34.5383	37.5866	35.998	21	3041
8	Worms	0.06	imbalanced	77.7778	13.4615	22.9508	2	43
			balanced	70.3704	36.5385	48.1013	0	33
9	Analysis	1.03	imbalanced	58.4906	11.5816	19.3347	56	654
			balanced	21.1356	16.6874	18.65	21	648
10	Backdoor	0.9	imbalanced	62.8099	10.8727	18.5366	5	618
			balanced	24.3976	11.588	15.7129	2	616
			imbalanced				1316	10884
			balanced				807	11913

2. The generic attack class ratio is around 22% for imbalanced DS showing the highest performance among all classes. The number of FN and FP are equal to 19 and 357, respectively, achieving a F1_Score of 98.8138%.
3. The normal data class is the second in performance values with FN equal to zero and FP equal to 1316 samples with a F1_Score of 95.2832%.
4. The Reconnaissance attack of imbalanced dataset has a class ratio of around 5% and is the third in performances rank, with FN equal to 19 and FP equal to 993 samples. F1_Score achieved a value of 82.8304% leading other types like: Exploits, Fuzzers and DoS, with ratios of 17.28%, 9.4% and 6.34%, respectively.
5. The Backdoor attack class of imbalanced dataset is with class ratio of 0.9% showing the least performance among all the classes. The number of FN and FP are equal to 5 and 618 samples, respectively.



a) UNSW_NB15 Imbalanced DS



b) UNSW_NB15 Imbalanced DS

Figure 9: Performance Metrics for Imbalanced and Balanced UNSW_NB15 DS

6. The technique SVMSOMTE is used to balance the DS. FN samples per class obtained from the RFC has decreased considerably from 1316 to 807 (shown in Table 6), which means implicitly an increase in the Recall metrics value. In the same time, it led to an increase in the false positive (FP) from 10,889 to 11,913, which means indirectly a reduction in the precision metric.
7. Performances of low-class numbers are improved by increasing the performance of Recall metrics, consequently the F1_Score was improved for all classes, as shown in Figure 9.
8. Performance metrics of RFC using UNSW_NB15 is shown graphically in Figure 10.

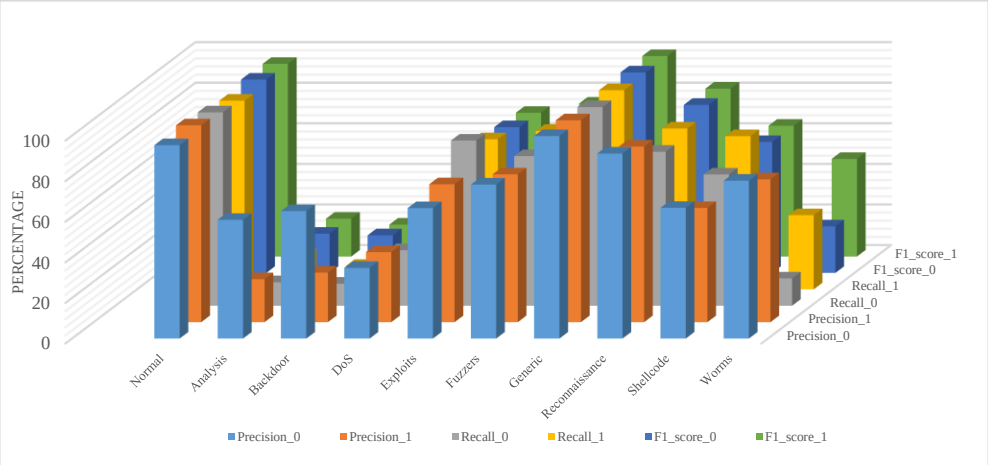


Figure 10: Performance Metrics for Imbalanced and Balanced UNSW_NB15 Dataset

Observations on CICIDS2017

The following points are key observations concerning CICIDS2017 dataset under RFC working on balanced and imbalanced datasets. Performance values supported by tabular data are shown in Table 7 and graphical analyses are shown in Figure 11 and Figure 12:

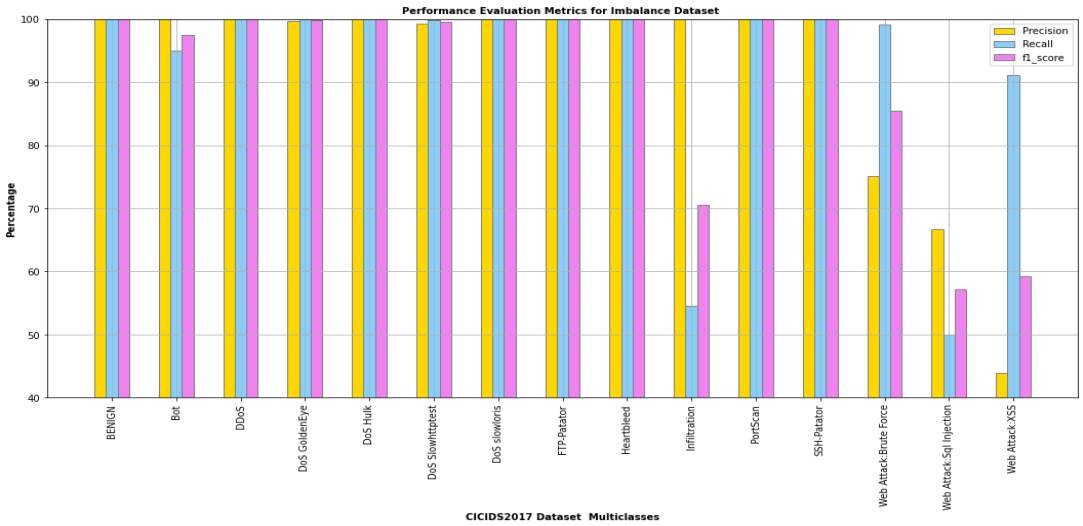
1. Overall accuracies of 99.8807% (for imbalanced) and 99.8613% (for balanced) are achieved.

Table 7: Performance Metrics for Imbalanced and Balanced CICIDS2017 Multiclass DS

No.	Class Name	Class Ratio%	Type	Precision%	Recall%	F1_score%	FN	FP
1	FTP-Patator	0.71	imbalanced	100	100	100	0	0
			balanced	99.958	100	99.979	0	0
2	Heartbleed	0.00089	imbalanced	100	100	100	0	0
			balanced	100	100	100	0	0
3	DDoS	11.47	imbalanced	100	99.9635	99.9818	13	1
			balanced	99.9974	99.9766	99.987	8	1
4	Port Scan	14.25	imbalanced	99.9853	99.9769	99.9811	2	9
			balanced	99.9937	99.979	99.9864	0	10
5	BENIGN	50	imbalanced	99.9731	99.942	99.9576	-	45
			balanced	99.9372	99.9683	99.9528	-	105
6	SSH-Patator	0.52	imbalanced	100	99.8869	99.9434	2	0
			balanced	100	99.9435	99.9717	1	0
7	DoS Hulk	20.71	imbalanced	99.912	99.9538	99.9329	25	7
			balanced	99.889	99.9437	99.9164	19	20
8	DoS slow loris	0.52	imbalanced	99.8843	99.3099	99.5963	2	10
			balanced	99.4203	98.6199	99.0185	1	23

9	DoS Golden Eye	0.92	imbalanced	99.7079	99.4819	99.5947	3	13
			balanced	99.1938	99.6114	99.4022	1	11
10	DoS Slow http test	0.49	imbalanced	99.2749	99.5758	99.4251	3	4
			balanced	97.214	99.3939	98.2919	3	7
11	Bot	0.17	imbalanced	100	94.9153	97.3913	30	0
			balanced	99.1511	98.9831	99.067	6	0
12	Web Attack Brute Force	0.13	imbalanced	75.0529	78.5398	76.7568	3	94
			balanced	74.2664	72.7876	73.5196	4	119
13	Infiltration	0.0032	imbalanced	100	54.5455	70.5882	5	0
			balanced	100	54.5455	70.5882	5	0
14	Web Attack Sql Injection	0.0018	imbalanced	66.6667	33.3333	44.4444	2	2
			balanced	60	50	54.5455	1	2
15	Web Attack XSS	0.058	imbalanced	43.9024	36.7347	40	7	117
			balanced	38.5366	40.3061	39.4015	4	113
			imbalanced				97	302
			balanced				53	411

2. Both attacks of FTP-Patator and Heartbleed that are with minor class ratios achieved values of 0.71% and 0.00089%, respectively, with zero FN and zero FP using imbalanced dataset.
3. The DDoS attack class has a class ratio of 11.47% achieving the third rank in the performance values, leading both the Port Scan and BENIGN classes with ratios of 14.25% and 50%, respectively.
4. SSH-Patator attack shows a ratio of 0.52%, surprisingly achieving a better performance than the DoS Hulk attack which comes with ratio of around 21%. They are sixth and seventh, respectively, as it was shown in Table 7.
5. The Web Attack: XSS class has the ratio of 0.058% with the lowest F1_Score value of 40.
6. The CICIDS2017 multiclass dataset is balanced with SMOTEENN technique, which is combination of SMOTE and the ENN (Edite Nearest Neighbours). It achieved an outstanding outcome in terms of reduced FN from 97 to 53 samples, which means implicitly an improvement of the Recall metric, especially for low ratio classes (as it was shown in Table 7). Performance metrics per class is shown graphically in Figure 11.



a) CICIDS2017 Imbalanced DS

b) CICIDS2017 Imbalanced DS

Figure 11: Performance Metrics for Imbalanced and Balanced CICIDS2017 DS

7. The performance metrics of RFC using CICIDS2017 balanced and imbalanced dataset is shown graphically by 3D in Figure 12.

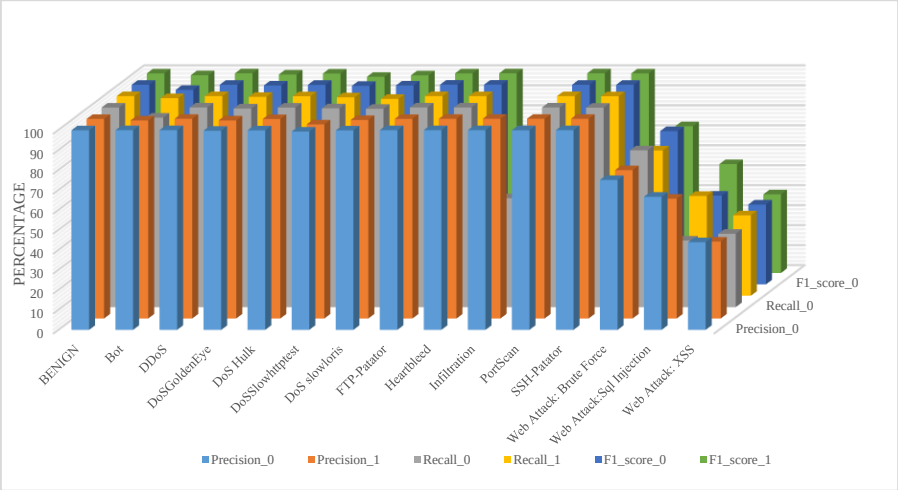


Figure 12: Performance Metrics for Imbalanced and Balanced CICIDS2017 DS

4 Comparisons and Conclusions

In this section comparisons and conclusions are presented highlighting key achievements in this work.

Comparisons

Multiclass classification for intrusion detection is the scope of this work through using three datasets (KDD99, CICIDS2017 and UNSW_NB15) with different data samples, input features, and number of output classes. The datasets are severely imbalanced meaning that the classifier used to train them has an excellent performance for major classes due to the well-trained parameters being obtained from enormous information available from major classes. In the same time, shortage of data instances from minor classes leads the classifier to have poor performance with minor classes. In real life, minor classes are the most important and crucial.

False negative outcomes of any classifier for minor classes are of a major importance. In the scope of this work, comparisons have been made between imbalanced and balanced data for each dataset. Different techniques were used to balance the datasets to improve performances for minority classes in all datasets in terms of precision, recall, and f1 score. Obtained outcomes for accuracy using KDD99 dataset have been compared with state-of-the-art related works as shown in Table 8.

Table 8: Accuracy Metric Comparison of this Work and others Using KDD99 DS

Type	**Dataset**	**This work**	(Keshta, 2018)	(Abrar et al., 2020)	(Mol & Mary, 2021)
Parameter	Used	Value %	Value %	Value %	Value %
Accuracy	KDD99	99.9852	99.84	99.48	99.6

Results obtained using KDD99 multiclass dataset in terms of Precision and Recall for all five classes are compared with a Reference (Sumaiya Thaseen et al., 2021) as in Table 9.

Table 9: Classes Comparison between this Work and others Using KDD99 DS

No.	**Class Name**	**Type**	**Precision%**	**Recall %**	**F1_score%**
1	DOS	Ref. (Sumaiya Thaseen et al., 2021)	74.41	74.41	
		This work	99.9991	99.9991	99.9991
2	normal	Ref. (Sumaiya Thaseen et al., 2021)	91.93	91.93	
		This work	99.9863	99.9486	99.9675
3	Probe	Ref. (Sumaiya Thaseen et al., 2021)	99.80	99.80	
		This work	99.9186	99.5942	99.7561
4	R2L	Ref. (Sumaiya Thaseen et al., 2021)	99.12	99.12	
		This work	98.7988	97.3373	98.0626
5	U2R	Ref. (Sumaiya Thaseen et al., 2021)	99.45	99.45	
		This work	92.8571	81.25	86.6667

In the same way, the results are compared with the Reference (Sumaiya Thaseen et al., 2021), using UNSW_NB15 dataset as shown in Table 10.

Table 10: 10 Classes Comparisons between this Work and others Using UNSW_NB15 DS

No.	**Class Name**	**Type**	**Precision%**	**Recall%**	**F1_score%**
1	Generic	Ref. (Sumaiya Thaseen et al., 2021)	99.87	97.33	
		This work	99.4719	98.1202	98.7914
2	Normal	Ref. (Sumaiya Thaseen et al., 2021)	99.98	100	
		This work	96.9836	93	94.95
3	Reconnaissance	Ref.(Sumaiya Thaseen et al., 2021)	99.40	66.61	
		This work	86.5055	79.2898	82.7406
4	Fuzzers	Ref. (Sumaiya Thaseen et al., 2021)	98.52	83.86	
		This work	72.8077	78.0726	75.3483
5	Exploits	Ref.(Sumaiya Thaseen et al., 2021)	88.11	89.44	
		This work	67.9458	73.9632	70.8269
6	Shellcode	Ref. (Sumaiya Thaseen et al., 2021)	99.81	36.51	
		This work	56.25	75.4967	64.4675
7	DoS	Ref. (Sumaiya Thaseen et al., 2021)	98.55	12.63	
		This work	34.5383	37.5866	35.998
8	Worms	Ref. (Sumaiya Thaseen et al., 2021)	100	24.64	
		This work	70.3704	36.5385	48.1013
9	Analysis	Ref. (Sumaiya Thaseen et al., 2021)	99.92	12.13	
		This work	21.1356	16.6874	18.65
10	Backdoor	Ref. (Sumaiya Thaseen et al., 2021)	100	63.94	
		This work	24.3976	11.588	15.7129

Also, the results are also compared with another Reference (Seth et al., 2021) for only seven classes, using CICIDS2017 dataset as shown in Table 11.

Table 11: 7 Classes Comparisons between this Work and others Using CICIDS2017 DS

No.	Class Name	Work	Precision%	Recall%	F1_score%
1	BENIGN	Ref. (Seth et al., 2021)	91.10	92.63	91.86
		This work	99.9372	99.9683	99.9528
2	Bot	Ref. (Seth et al., 2021)	99.91	99.63	99.77
		This work	99.1511	98.9831	99.067
3	Brute Force	Ref. (Seth et al., 2021)	72.78	99.94	84.23
		This work	74.2664	72.7876	73.5196
4	DDoS	Ref. (Seth et al., 2021)	98.62	99.84	99.23
		This work	99.9974	99.9766	99.987
5	DoS	Ref. (Seth et al., 2021)	99.93	77.33	87.19
		This work	99.889	99.9437	99.9164
6	Infiltration	Ref. (Seth et al., 2021)	45.91	31.87	37.62
		This work	100	54.5455	70.5882
7	Web Attack	Ref. (Seth et al., 2021)	90.37	98.81	94.40
		This work	66.6667	33.3333	44.4444

Finally, the results of this work using CICIDS2017 dataset with fifteen (15) classes are compared with Reference (Ferrag et al., 2020) and shown in Table 12.

Table 12: 15 Classes Comparisons between this Work and others Using CICIDS2017 DS

No.	Class Name	Type	Precision%	Recall%	F1_score%
1	FTP-Patator	Ref. (Ferrag et al., 2020)		99.636	
		This work	99.958	100	99.979
2	Heartbleed	Ref. (Ferrag et al., 2020)		100	
		This work	100	100	100
3	DDoS	Ref. (Ferrag et al., 2020)		99.879	
		This work	99.9974	99.9766	99.987
4	Port Scan	Ref. (Ferrag et al., 2020)		99.881	
		This work	99.9937	99.979	99.9864
5	BENIGN	Ref. (Ferrag et al., 2020)		98.855	
		This work	99.9372	99.9683	99.9528
6	SSH-Patator	Ref. (Ferrag et al., 2020)		99.909	
		This work	100	99.9435	99.9717
7	DoS Hulk	Ref. (Ferrag et al., 2020)		96.782	
		This work	99.889	99.9437	99.9164
8	DoS slow loris	Ref. (Ferrag et al., 2020)		97.758	
		This work	99.4203	98.6199	99.0185
9	DoS Golden Eye	Ref. (Ferrag et al., 2020)		67.571	
		This work	99.1938	99.6114	99.4022
10	DoS Slow http test	Ref. (Ferrag et al., 2020)		93.841	
		This work	97.214	99.3939	98.2919
11	Bot	Ref. (Ferrag et al., 2020)		46.474	
		This work	99.1511	98.9831	99.067
12	Web Attack Brute Force	Ref. (Ferrag et al., 2020)		73.265	
		This work	74.2664	72.7876	73.5196
13	Infiltration	Ref. (Ferrag et al., 2020)		100	
		This work	100	54.5455	70.5882
14	Web Attack Sql Injection	Ref. (Ferrag et al., 2020)		50	
		This work	60	50	54.5455
15	Web Attack XSS	(Ferrag et al., 2020)		30.625	
		This work	38.5366	40.3061	39.4015

Conclusions

Key points inferred from outcomes of this work are as follows:

- The two methods suggested and used in this work (Tomek and Smote + ENN) enhanced the performance and outcome values for positive minority instances.
- The KDD99 dataset is balanced by combining oversampling (represented by SMOTE) and under-sampling (represented by Tomek link method) techniques. With the KDD99 dataset, the Random Forest classifier improved in performances and reduction in false negative results from 23 to 15. The lowest class ratios reached for U2R 0.01% and R2L reached 0.22%. An increase in false positive results is noticed by low number classes duplications, particularly those that are located in major class overlap regions. The number of false positives instances rose from 4 to 7.
- The UNSW_NB15 dataset is balanced using a combination of SMOTE and SVM algorithms to be classified by Random Forest. The model achieved an out-standing performance improvement in terms of false negative rates of around 38.68%, but at the same time (unfortunately) the false positive prediction in-creased to be around 9.45%.
- The CICIDS2017 dataset is balanced by using a combination of SMOTE and ENN techniques. The Random Forest classifier achieved a significant improve-ment in terms of false negative rates by around 45.36%, but at the same time the drawback false positive rate increased to be around 36.09%.

References

[1] Abdulhammed, R., Musafer, H., Alessa, A., Faezipour, M., & Abuzneid, A. (2019). Features dimensionality reduction approaches for machine learning based network intrusion detection. *Electronics*, *8*(3), 1-27.

[2] Abrar, I., Ayub, Z., Masoodi, F., & Bamhdi, A.M. (2020). A machine learning approach for intrusion detection system on NSL-KDD dataset. *2020 International Conference on Smart Electronics and Communication (ICOSEC)*, 919–924.

[3] Ahmad, Z., Shahid Khan, A., Wai Shiang, C., Abdullah, J., & Ahmad, F. (2021). Network intrusion detection system: A systematic study of machine learning and deep learning approaches. *Transactions on Emerging Telecommunications Technologies*, *32*(1), 10-1002.

[4] Al-Daweri, M. S., Zainol Ariffin, K. A., Abdullah, S., & Md. Senan, M. F. E. (2020). An analysis of the KDD99 and UNSW-NB15 datasets for the intrusion detection system. *Symmetry*, *12*(10), 1-32.

[5] Almseidin, M., Alzubi, M., Kovacs, S., & Alkasassbeh, M. (2017). Evaluation of machine learning algorithms for intrusion detection system. *2017 IEEE 15th International Symposium on Intelligent Systems and Informatics (SISY)*, 277–282.

[6] Belouch, M., El Hadaj, S., & Idhammad, M. (2018). Performance evaluation of intrusion detection based on machine learning using Apache Spark. *Procedia Computer Science*, *127*, 1-6.

[7] Bertoli, G.D.C., Júnior, L.A.P., Saotome, O., Dos Santos, A.L., Verri, F.A.N., Marcondes, C.A.C., Barbieri, S., Rodrigues, M.S., & De Oliveira, J.M.P. (2021). An end-to-end framework for machine learning-based network intrusion detection system. *IEEE Access*, *9*, 106790–106805.

[8] bhai Gupta, A.R., & Agrawal, J. (2020). A comprehensive survey on various machine learning methods used for intrusion detection system. *IEEE 9th International Conference on Communication Systems and Network Technologies (CSNT)*, 282–289.

[9] Brownlee, J. (2020). *Imbalanced classification with Python: better metrics, balance skewed*

classes, cost-sensitive learning. Machine Learning Mastery.
[10] Daniya, T., Kumar, K.S., Kumar, B.S., & Kolli, C.S. (2021). *WITHDRAWN: A survey on anomaly based intrusion detection system*. Elsevier.
[11] Farhana, K., Rahman, M., & Ahmed, M. (2020). An intrusion detection system for packet and flow based networks using deep neural network approach. *International Journal of Electrical & Computer Engineering (2088-8708)*, *10*(5), 5514-5525.
[12] Ferrag, M.A., Maglaras, L., Ahmim, A., Derdour, M., & Janicke, H. (2020). Rdtids: Rules and decision tree-based intrusion detection system for internet-of-things networks. *Future Internet*, *12*(3), 1-14.
[13] Fitni, Q.R.S., & Ramli, K. (2020). Implementation of ensemble learning and feature selection for performance improvements in anomaly-based intrusion detection systems. *2020 IEEE International Conference on Industry 4.0, Artificial Intelligence, and Communications Technology (IAICT)*, 118–124.
[14] Galar, M., Fernandez, A., Barrenechea, E., Bustince, H., & Herrera, F. (2011). A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, *42*(4), 463–484.
[15] Hooshmand, M.K., & Gad, I. (2020). Feature selection approach using ensemble learning for network anomaly detection. *CAAI Transactions on Intelligence Technology*, *5*(4), 283–293.
[16] Hussein, M.A. (2022). Performance Analysis of different Machine Learning Models for Intrusion Detection Systems. *Journal of Engineering*, *28*(5), 61–91.
[17] Iman, A.N., & Ahmad, T. (2020). Improving intrusion detection system by estimating parameters of random forest in Boruta. *International Conference on Smart Technology and Applications (ICoSTA)*, 1–6.
[18] Jain, A., & Rana, J.L. (2016). Classifier selection models for intrusion detection system (IDS). *Informatics Engineering, an International Journal (IEIJ)*, *4*(1), 1–11.
[19] Keshta, I.M. (2018). Intelligent intrusion detection system based on MLP, RBF and SVM classification algorithms: a comparative study. *International Journal of Computer Science and Information Security*, *16*(5).
[20] Krawczyk, B. (2016). Learning from imbalanced data: open challenges and future directions. *Progress in Artificial Intelligence*, *5*(4), 221–232.
[21] Li, G., Yan, Z., Fu, Y., & Chen, H. (2018). Data fusion for network intrusion detection: a review. *Security and Communication Networks*, *2018*.
[22] Liloja & Dr. Ranjana, P. (2023). An Intrusion Detection System Using a Machine Learning Approach in IOT-based Smart Cities. *Journal of Internet Services and Information Security (JISIS)*, *13*(1), 11-21.
[23] Liu, C., Gu, Z., & Wang, J. (2021). A hybrid intrusion detection system based on scalable K-Means+ random forest and deep learning. *IEEE Access*, *9*, 75729–75740.
[24] Mebawondu, J.O., Alowolodu, O.D., Mebawondu, J.O., & Adetunmbi, A.O. (2020). Network intrusion detection system using supervised learning paradigm. *Scientific African*, *9*.
[25] Meryem, A., & Ouahidi, B.E.L. (2020). Hybrid intrusion detection system using machine learning. *Network Security*, *2020*(5), 8–19.
[26] Mishra, P., Varadharajan, V., Tupakula, U., & Pilli, E.S. (2018). A detailed investigation and analysis of using machine learning techniques for intrusion detection. *IEEE Communications Surveys & Tutorials*, *21*(1), 686–728.
[27] Mol, P.R., & Mary, C.I. (2021). Classification of Network Intrusion Attacks Using Machine Learning and Deep Learning^. *Annals of the Romanian Society for Cell Biology*, 1927–1943.
[28] Ravipati, R.D., & Abualkibash, M. (2019). Intrusion detection system classification using different machine learning algorithms on KDD-99 and NSL-KDD datasets-a review paper. *International Journal of Computer Science & Information Technology (IJCSIT)*, *11*(3), 65-80.

[29] Salih, A.A., & Abdulazeez, A.M. (2021). Evaluation of classification algorithms for intrusion detection system: A review. *Journal of Soft Computing and Data Mining*, *2*(1), 31–40.
[30] Sandosh, S., Govindasamy, V., & Akila, G. (2020). Enhanced intrusion detection system via agent clustering and classification based on outlier detection. *Peer-to-Peer Networking and Applications*, *13*, 1038–1045.
[31] Seth, S., Chahal, K.K., & Singh, G. (2021). A novel ensemble framework for an intelligent intrusion detection system. *IEEE Access*, *9*, 138451–138467.
[32] Subba, B., & Gupta, P. (2021). A tfidfvectorizer and singular value decomposition based host intrusion detection system framework for detecting anomalous system processes. *Computers & Security*, *100*.
[33] Sumaiya Thaseen, I., Saira Banu, J., Lavanya, K., Rukunuddin Ghalib, M., & Abhishek, K. (2021). An integrated intrusion detection system using correlation-based attribute selection and artificial neural network. *Transactions on Emerging Telecommunications Technologies*, *32*(2).
[34] Wahba, Y., ElSalamouny, E., & ElTaweel, G. (2015). Improving the performance of multi-class intrusion detection systems using feature reduction, *12*(3), 255-262.
[35] Xin, Y., Kong, L., Liu, Z., Chen, Y., Li, Y., Zhu, H., Gao, M., Hou, H., & Wang, C. (2018). Machine learning and deep learning methods for cybersecurity. *IEEE Access*, *6*, 35365–35381.
[36] Zhang, B., Liu, Z., Jia, Y., Ren, J., & Zhao, X. (2018). Network intrusion detection method based on PCA and Bayes algorithm. *Security and Communication Networks*, *2018*, 1–11.
[37] Zhou, Y., Cheng, G., Jiang, S., & Dai, M. (2020). Building an efficient intrusion detection system based on feature selection and ensemble classifier. *Computer Networks*, *174*, 1-21.
[38] Zhu, H., Liu, W., Sun, M., & Xin, Y. (2017). A universal high-performance correlation analysis detection model and algorithm for network intrusion detection system. *Mathematical Problems in Engineering*, *2017*.

Authors Biography

Salim Q. Mohammed received his BSc degree in Electrical Engineering from Baghdad University. He received his MSc in Electronic and Communication Engineering from The University of Sulaimani. He is currently a PhD. student and lecturer at Sulaimani Polytechnic University. His research interests are in the fields of Artificial Intelligence, Machine Learning, Communications, and Electronics hardware implementations.

Mohammed Hussein, graduated from Control and System Engineering department in 1986 from University of Technology in Baghdad and completed his Master in Computer Engineering in 1991, and obtained his PhD in Computer Science in 2007. Since his graduation date, he has conducted many researches works and published 19 papers in the field of computer control, intrusion detection and machine learning. His current interest is AI application in engineering fields. Through his job career, he had several positions, such as the head of computer science department and the dean of technical College of Informatics. Currently, he is an associate professor at communication engineering department of Sulaimani Polytechnic University.