Securing Multi-Environment Networks using Versatile Synthetic Data Augmentation Technique and Machine Learning Algorithms

Furqan Rustam
and Anca Delia Jurcut
School of Computer Science
University College Dublin
D04 V1W8, Ireland
Email: furqan.rustam1@gmail.com
and anca.jurcut@ucd.ie

Wajdi Aljedaani
University of North Texas
Denton, TX 76205, USA
Email:wajdi.j1@gmail.com

Imran Ashraf
Department of Information
and Communication Engineering
Yeungnam University
Gyeongsan, 38541, Korea
Email:imranashraf@ynu.ac.kr

2023 20th Annual International Conference on Privacy, Security and Trust (PST) | 979-8-3503-1387-1/23/$31.00 ©2023 IEEE | DOI: 10.1109/PST58708.2023.10320203

***Abstract*—The emergence of new network architectures, protocols, and tools has made it easier for cybercriminals to launch attacks using AI-based tools, presenting challenges in network security. To protect such systems, a versatile malicious traffic detection system is required that can identify attacks regardless of the type of traffic coming toward the network. In this paper, a system is proposed that can singly analyze multi-environment traffic (IoT and traditional IP-based) to detect malicious activity. The existing techniques for managing Multi-Environment traffic are inefficient due to the absence of AI utilization. To overcome these issues, the proposed approach generates a novel multi-environment traffic dataset by merging existing network datasets containing both traditional IP-based traffic and IoT network traffic. Synthetic Data Augmentation TEchnique (S-DATE) is also proposed to overcome the problem of imbalanced data distribution in the new multi-environment dataset. The results show that the utilization of S-DATE results in faster machine learning model convergence and an improvement in the detection rate of normal and abnormal traffic. The proposed approach achieves an impressive overall detection rate of 0.991 and is statistically significant compared to other state-of-the-art approaches.**

I. Introduction

Cyberattacks have numerous forms, including malware, denial of service (DoS) attacks, ransomware, and phishing [1]. Such attacks has malicious intent and often target personal information, stealing financial information, confidential and sensitive data, and pin and card numbers, etc. [2]. Cybercriminals follow different suites to carry out these attacks and use different methods and resources including insider threats, software or hardware vulnerabilities, and social engineering [3]. The frequency of cyberattacks is increasing day by day because of several reasons. First is the increased reliance on technology in our daily life which is attack prone. Secondly, the proliferation of the Internet of Things (IoT), which connects devices through the Internet and increases the chance of hacking home devices, industrial machines, and healthcare systems which may lead to significant disruption and damage [4]. Third, cyber-attacks are becoming more sophisticated and organized, employing techniques and tactics using the latest tools. The attackers are using artificial intelligence-based methods to breach the security system easily. Besides that, COVID-19 shifted working from remote locations thus creating more vulnerabilities for cyber criminals to exploit [5]. According to [6], there is a 38% increase in cyber attacks during 2022 as compared to 2021. Another report highlights the destruction of cyber attacks and approximately 236.1 million ransomware attacks occurred worldwide, only during the first half of 2022 [7].

The increase in the frequency and severity of cyberattacks demands an urgent need for robust security mechanisms to protect personnel, business, and governments from cyber criminals. Many researchers are working on identifying effective strategies for mitigating cyber threats such as encryption-decryption techniques [8], firewalls [9], and intrusion detection systems [10] as well as non-technical measures such as cyber security awareness, training of employees, and risk assessment. The current network anomaly detection systems face significant challenges due to the evolution of network architecture and the rapid expansion of network equipment [11]. Traditional networks now include IoT devices because of their wide use in various fields. IoT devices are a revelation in the future for the industry as well as in personal networks. Still, the security mechanism of IoT devices is poor, which provides limited security mechanisms and puts the whole network at risk. A vast majority of these devices do not have any security measures, which leaves them vulnerable to internet exposure, making them susceptible to thousands of vulnerabilities that can be exploited by network attacks [11]. Therefore, in future network technologies, the network's security issue becomes a key focus, especially the hybrid networks where both the traditional IP and IoT network traffic exist. In [12], the authors raised the concept of multi-environment network traffic and proposed a machine learning approach for malicious traffic detection from multi-environment traffic data. There is still a big gap in this domain as no dataset is publicly available for hybrid network traffic, and the performance with existing

datasets is also poor, as reported in [12].

The existing techniques available for handling multi-environment networks include physical network segmentation or virtual local area network (VLAN) [13]. However, these methods have their drawbacks. Network segmentation increases the cost of networks, and traffic entering from a single source still poses a significant concern. VLANs, on the other hand, may result in packet leakage, which elevates the risk of compromising network security. Additionally, a threat within a single system may propagate a virus through an entire logical network. In response to above-mentioned problems and limitations in multi-environment network traffic, this study proposed an approach using artificial intelligence approach so that malicious traffic can be handled at the top level. This study makes several contributions concerning malicious traffic detection

- Novel multi-environment (M-En) traffic data is generated by utilizing existing datasets (UNSW-NB15, IoTID20) that contain both traditional IP-based traffic and IoT network traffic. The datasets are merged and divided into two categories, namely anomaly and normal. The hybrid behavior of the dataset, which includes both IoT and traditional IP traffic samples, enables the training of a single model for both types of traffic.
- Synthetic Data Augmentation TEchnique (S-DATE) is proposed that generates synthesized data. This technique helps to overcome the new M-En dataset problem. New generated M-En dataset consists of imbalanced data distribution as samples for the normal class are low in number as compared to anomaly class data which somehow can cause model over-fitting on majority class data. The proposed S-DATE generates more sample for normal class data by using the existing samples. New generated samples by the S-DATE will be more correlated with existing samples which helps to achieve significant accuracy.
- Extensive analysis is performed to address the issue of low detection rates and high false alarm rates for normal samples. To accomplish this, several data balancing techniques, including the synthetic minority oversampling technique (SMOTE), adaptive synthetic (ADASYN), and conditional GAN (CTGAN) are used in comparison with the proposed S-DATE method. The utilization of S-DATE results in faster model convergence and an improvement in the detection rate of normal traffic.
- We experiment on several types of datasets as well as the M-En dataset. The performance of the proposed S-DATE is significant as with an impressive overall detection rate of 0.991. The proposed method has shown its adaptability and appropriateness for identifying malicious traffic in M-En.

This paper is further divided into six sections. Related work is presented in Section II while synthetic data augmentation techniques are given in Section III. The M-En dataset is discussed in Section IV while the proposed methodology is given in Section V. Section VI presents experimental results and discusses, followed by the conclusion in Section VII.

II. Related Work

Over the past decade, the field of computer networks has seen rapid technological advancements, including the development of various tools, protocols, and architectures. This has made the task of securing networks against cybercriminals increasingly challenging [24], [25]. In this section, we examine recent research conducted on the detection of malicious traffic.

A. Intrusion Detection for Traditional IP-based Traffic

Several studies work with traditional IP traffic datasets such as UNSW-NB15, KDD-99, CICIDS2017, CICIDS2018, and many more [26]–[28]. The study [14] proposed a hybrid method for detecting network intrusions that are not dependent on any particular dataset. The authors proposed a hybrid model BLoCNet, which is a combination of convolutional neural networks (CNN) and bidirectional long short-term memory (Bi-LSTM). Three datasets, CIC-IDS2017, IoT-23, and UNSW-NB15 are used to perform experiments with BLoCNet. The proposed BLoCNet model achieved 98%, 99%, and 76.34% accuracy scores for CIC-IDS2017, IoT-23, and UNSW-NB15 datasets, respectively. The study [15] used machine learning and deep learning approaches for cyber security. Several machine learning models, such as random forest (RF), Adaboost, XGBoost, K nearest neighbour (KNN) and a deep learning model deep multi-layer perceptron (deep MLP) are used to perform experiments on the UNSW-NB15 dataset. The performance of RF is significant, with a 98.96% accuracy score for network attack detection. Early prediction of attacks in network security is a need of time, and many researchers are designing approaches in this regard. For example, [16] proposed an approach for early warning of network anomalies. The authors used generalized network temperature (GNT) and deep learning model Bi-gated recurrent unit (Bi-GRU). The proposed Bi-GRU first classifies the DDoS-induced network congestion. Two datasets, CICIDS2017 and UNSW-NB15, are used to perform experiments. Bi-GRU achieved 96.84% and 95.68% accuracy on the CICIDS2017 and UNSW-NB15 datasets, respectively.

B. Intrusion Detection for IoT-based Datasets

Numerous studies have also focused on analyzing IoT traffic datasets, including IoTID-20 and CIC IoT 2022 [29], [30]. In [19], a new method for detecting intrusions in IoT networks was presented. The proposed approach called the ShieldRNN model, was tested on the CIC-IDS2017 and CIC IoT 2022 datasets, resulting in an impressive 99.989% accuracy for the CIC IoT 2022 dataset. In their research paper [20], the authors utilized machine learning and deep learning techniques to enhance the security of IoT networks. To improve the accuracy of attack detection, they employed feature engineering methods, including principal component analysis (PCA), linear discriminant analysis (LDA), and Auto-encoder in combination with learning models. Their proposed approach was evaluated

TABLE I: Summary of related work.

Ref	Year	Approach	Dataset	Type	Method
[14]	2023	DL	CIC-IDS2017, IoT-23 and UNSW-NB15	Tr.IP	CNN and BiLSTM, and BLoCNet are used for network intrusion detection.
[15]	2023	DL, ML	UNSW-NB15	Tr.IP	RF, Adaboost, XGBoost, KNN, and deep MLP are used for network attack detection.
[16]	2023	DL	CICIDS2017, UNSW-NB15	Tr.IP	Hybrid stack model Bi-GRU is used for early warning prediction of network anomalies.
[17]	2023	DL	UNSW-NB15	Tr.IP	Bi-LSTM and LSTM+CNN with Mayfly optimizer to detect the malware from network traffic.
[18]	2023	SSML, DL	UNSW-NB15	Tr.IP	SSMlL approach WGAN-GP for malicious traffic detection.
[19]	2022	DL	CIC-IDS2017, CIC IoT 2022	IoT, Tr.IP	Proposed RNN-based model for intrusion detection in IoT networks.
[20]	2022	DL, ML	ToN-IoT, CSE-CIC-IDS2018, UNSW-NB15	IoT, Tr.IP	Feature extraction approach with ML and DL model for intrusion detection in IoT networks.
[21]	2022	DL	TON IoT, NSL-KDD, CIC-IDS2018	IoT, Tr.IP	Hybrid model for network intrusion detection.
[22]	2022	DL	NF-UNSW-NB15, NF-ToN-IoT, NF-BoT-IoT, NF-CSE-CIC-IDS2018	IIoT, IoT, Tr.IP	Proposed neural network based approach for IIoT networks intrusion detection.
[23]	2022	ML	Darknet	IoT	Ensemble approach intrusion detection in IoT dataset.

using various datasets such as ToN-IoT, CSE-CIC-IDS2018, and UNSW-NB15. The results of their experiments revealed that decision tree (DT) with Auto-encoder features achieved an accuracy of 98.23% for the ToN-IoT dataset. In contrast, logistic regression (LR) with full features achieved an accuracy of 99.27% for the UNSW-NB15 dataset. Additionally, for the CSE-CIC-IDS2018 dataset, DT with all features achieved an accuracy of 98.15%.

C. Gap and Summary of Related Work

We summarized the recent literature in the network security domain using machine learning in Table I, which is discussed above. Researchers used different dataset IoT and traditional IP-based (Tr.IP) traffic datasets. While in machine learning approaches, researchers consider ensemble models and feature engineering techniques to detect abnormal traffic with high accuracy. Recent research did not focus on the new architecture of networks such as M-En, where both IoT and Tr.IP traffic flow under single networks. The traditional approach to tackle the M-En networks, including physical network segmentation or VLAN, is not secure with their limitations which we discussed in Section I. There is no study that can be seen in the literature working with the imbalanced data problem, which is present in many existing networks dataset. This research work focused on M-En traffic and tried to overcome our previous research limitations in the M-En domain [12] using existing datasets and novel data balancing techniques.

III. Synthetic Data Augmentation Technique

We proposed a synthetic data augmentation approach to deal with the imbalanced dataset problem. Our technique generates new data by modifying the copies of existing data which is more correlated to the existing data. We work on feature space and try to generate new data more compact to existing data rather than scattering on large space. We are using the concept of nearest neighbors in this technique [31], but to find the nearest neighbor, we work on the outputs by the dimensionality reduction techniques. Only one neighbor is considered to generate a new sample by calculation with existing copies. Other approaches, such as SMOTE and ADASYN, also work based on the nearest neighbor [32]. To create randomness in data, they multiply output with a random number between 0 and 1 while we work on only actual data and generate new samples, which will be very close to the original sample with distinguishable values [33].

The S-DATE algorithm is the pseudo-code for the proposed approach. The proposed approach needs input data in numerical form, and then it reduces the feature space into 2D using singular vector decomposition. Finding the neighbors based on multiple features is more complex as compared to 2D space, and it also helps to reduce the computation cost of proposed algorithms. This dimensionality reduction of the data before applying the next steps can help to reduce the impact of noise or irrelevant features on the synthetic sample. S-DATE is flexible to adopt any kind of technique for feature reduction, such as principle component analysis, chi-square, or any other neural networks-based technique, because results can vary from data to data. After reducing the dimensions of feature space, we compute the distance of a sample with all other samples using Euclidean distance. It is flexible in the proposed approach to using any kind of distance matrix according to the nature dataset. We find the nearest samples based on distance and take that samples and the original sample to compute the means of both samples' feature values. The mean of each feature value is a new synthetic sample. To avoid data leakage or duplication, we consider each sample only once to compute the mean with the nearest neighbor.

Figure 1 illustrate all the steps in the S-DATE technique. S-DATE consists of 4 steps, 1- Concerting data into 2D space, 2- Measuring the distance of the sample with all other samples (i.e., $d_1, d_2, d_3, ..., d_n$), 3- Find the number of samples with minimum distance (S_{md}) from a sample and calculate the

mean of samples feature-wise to create a synthetic sample (SS) and continue till the required number of samples are generated, 4- If the number of required samples are more than the number of original samples than start again from step 1. The algorithm 1 also illustrates the pseudocode for the novel S-DATE.

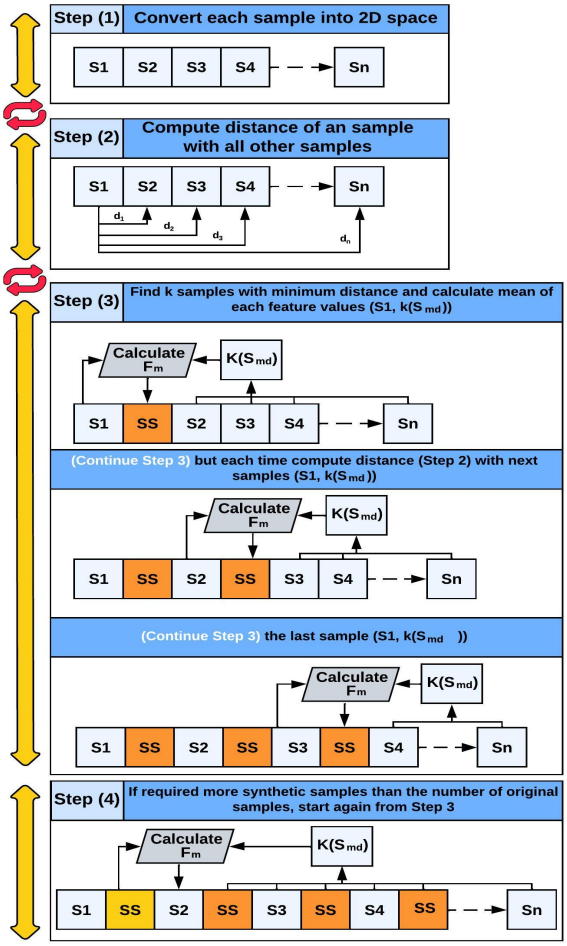
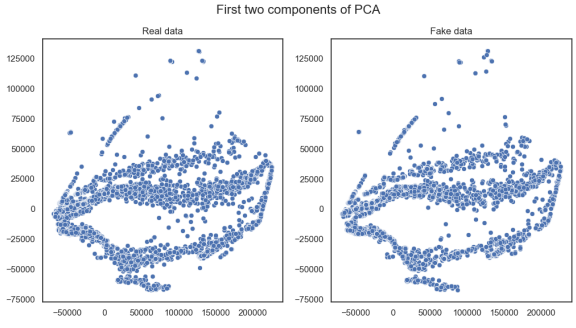


Fig. 1: S-DATA processing steps

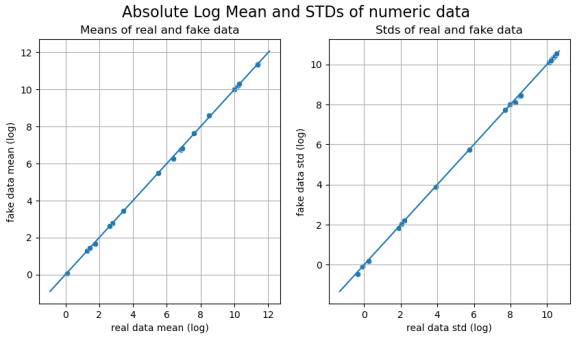
Figures 2 show the similarity between the S-DATE synthesized and real data. We used the TableEvaluator library to perform the evaluation. Figure 2a illustrates the fake and real data in 2D space, and they look very similar, which shows that the generated data and actual data are very correlated, which will help the model to learn similar kinds of patterns either from real or fake data. Figures 2b show the difference between real and fake data.

IV. M-En Dataset Description

AI-based network security system strength is based on the used dataset for the training of models that researchers are generating new datasets by considering the latest architecture of networks. Our research focused on M-En networks, but there is no publicly available dataset on such networks. To overcome this problem, we used two datasets, IoT-based traffic (IoTID-20 [29]) and Tr. IP-based traffic (UNSW-NB15 [26])



(a)



(b)

Fig. 2: Similarity between synthesized data and real data.

datasets. We combine these datasets to generate the M-En traffic dataset.

To generate the M-En dataset, we combine both datasets class by class, but the problem with datasets is that they are unequal in the number of features. To generate a new dataset with impactful features first, we extract important features using an extra trees classifier (EXTC). We pass each dataset feature to EXTC with the corresponding target. EXTC finds how each feature helps to categorize data into their corresponding target and assigns a score to each feature. We select the top 20 impact features from both datasets and then combine both datasets and divide them into Normal and Anomaly categories. Table II shows the count for the M-En dataset.

TABLE II: Dataset count for each category

Categories	IoTID-20	UNSW-NB15	M-En
Anomaly	585710	164673	750383
Normal	40073	93000	133073

V. Proposed Methodology

This section discusses the data flow in the proposed approach, as depicted in Figure 3. Initially, we obtained two network traffic datasets with distinct traffic patterns, namely IoT and general traffic. These datasets were combined to generate the M-En traffic dataset, as explained in Section IV. Afterward, we shuffled and mixed the dataset to reduce bias. However, the original M-En dataset size was too large, which could lead to high computational costs. It contained

Algorithm 1 S-DATE algorithm

Input: S-DATE(*S, N,* D_r*,* D_m*, k*)
    S = Original data
    N = Number of synthetic samples
    D_r = Object for dimensional reduction technique ▷ i. e. PCA, SVD, ... Note: This study used SVD
    D_m = Object for distance matrix technique ▷ i. e. Euclidean
    k = Number of neighbours with minimum distance ▷ Note: k=10
Output: Synthetic data= *size(S+N)*
$2D_d = D_r(S)$ ▷ Converting data in 2D space
while $m \neq N$ **do**
    Dis[m+1]=D_m(2D_d*[m:m+1]*,2D_d*[:]*).reshape(-1) ▷ D_m will find the distance *Dis* of each sample with all next samples in the dataset (Note: This step will be performed on 2D data)
    $k(S_{md})$ = *S*[*Dis.idxmin()*] ▷ Find k samples with minimum distance S_{md} next to *S*[m].
    SS = ($k(S_{md})$, *S*[m]).mean(axis=0) ▷ Generate synthetic samples (SS) by taking mean of S_{md} and *S*[m] in feature space rather than data space.
    if $m > len(S)$ **then** ▷ If the number of synthetic required more than the number of the original samples, take more iterations and this time include already generated SS to generate new SS .
        Goto (Start from step 1)
        N=len(S)-N
    else if $m == N$ **then**
        return (shuffle(*S+SS*)) ▷ If the required number of samples are generated, then combine original samples and SS and shuffle them. Note: Shuffle is of optional
    end ifm=m+1
end while

a total of 883,456 samples, comprising 133,073 'Normal' and 750,383 'Anomaly' samples. To conduct our experiments, we used only 25% of the dataset, which included 76,520 'Normal' and 144,344 'Anomaly' samples, resulting in an imbalanced dataset. To address this issue, we applied the S-DATE approach, which synthesizes data for the 'Normal' class to balance both target classes. The resulting balanced dataset consisted of 288,688 samples, comprising 144,344 'Normal' and 144,344 'Anomaly' samples.

Figure 4 presents two approaches for implementing the S-DATE technique. The first approach involves applying S-DATE before data splitting, while the second approach involves deploying S-DATE on the training set after data splitting. In the first approach, the dataset undergoes data balancing using S-DATE, followed by the division of the dataset into training and testing sets. However, this approach raises concerns regarding data leakage and potential model overfitting, as applying data balancing before splitting may lead to information leakage [34], [35]. Alternatively, in the second approach, S-DATE is exclusively applied to the training set, mitigating the risk of data leakage and ensuring a more robust model evaluation process. We split the dataset into training and testing sets in a ratio of 85:15, where 85% of the data was used for model training and 15% for model testing.

Subsequently, we applied multiple models, including RF, ETC, ADA, and LR. For each model, we identified the best hyperparameter settings through a hyperparameter tuning process, where we varied the hyperparameter values within a specific range, as presented in Table III. Once the models were deployed, we evaluated their performance in terms of accuracy,

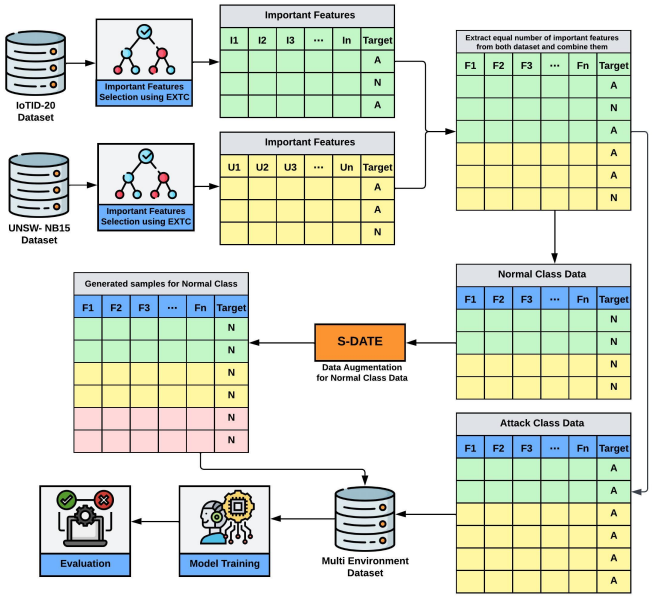


Fig. 3: Proposed methodology diagram

precision, recall, F1 score, number of correct predictions (CP), and number of wrong predictions (WP).

VI. Results and Discussion

Experiments of this study are performed on Core i7 11th generation machine with 64GB RAM and Integrated Intel Iris Xe Graphics. We used Jupyter Notebook and Python

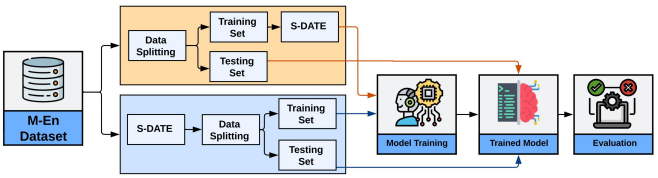


Fig. 4: S-DATE deploying approaches

TABLE III: Models hyperparameters setting

Model	Hyperparamets	Tuning Range
RF	n_estimators = 300, max_depth =100	n_estimators = {20 to 400}, max_depth ={5 to 200}
ETC	n_estimators = 300, max_depth =100	n_estimators = {20 to 400}, max_depth ={5 to 200}
ADA	n_estimators = 300, learning_rate =0.8	n_estimators = {20 to 400}, learning_rate ={0.1 to 1.0}
LR	solver = 'liblinear', multi_class ='ovr'	solver = ['liblinear', 'sag', 'saga'], multi_class = ['ovr', 'multinomial']

with several libraries, including sci-kit Learn, TensorFlow, and Keras.

A. Results for M-En Dataset

Table IV shows the results of a machine learning model on the M-En dataset, which has an original target class distribution. The RF model shows a good accuracy score of 0.985, which is considered satisfactory in machine learning but not in the context of cyber security. Even a small error margin of 0.01% can pose a significant threat. On the other hand, other models, such as ETC, ADA, and LR, perform poorly in terms of accuracy due to the imbalanced distribution of the dataset. The minority class, which is the Normal class in this case, has lower accuracy than the majority class due to its imbalanced representation in the dataset. Specifically, the RF model makes 378 incorrect predictions for the Normal class and 115 for the Anomaly class. In comparison, the ETC, ADA, and LR models produce 654, 398, and 2464 incorrect predictions for the Normal class, respectively, while for the Anomaly class, these figures are much lower at 270, 160, and 1333, respectively. Normal class wrong prediction is high even though the number of sample ratio for Normal class in testing data is low as compared to Anomaly class which totally models over-fitting towards the majority class. This imbalance in the dataset leads to the overfitting of the majority class and underfitting of the minority class, thereby reducing the overall accuracy of the models. To address this issue, we employed data balancing techniques to improve the accuracy of the models in predicting network traffic.

S-DATE help to resolve the class distribution by generating synthesized data for the Normal class. S-DATE generates more correlated and is very near to original samples because of its simple mathematical approach, as we mention in Section III. After deploying the S-DATE approach we trained learning models on a balanced dataset. The results are significant as the model achieved 0.991 accuracy scores with balancing accuracy for both target classes. The other models are also good with balancing techniques as the fluctuation between precision and recall is not too high, as shown in Table V. The wrong prediction for the Normal class is 222, and for Anomaly is 181, which shows that the model is performing well in both classes and there is not a big difference.

TABLE IV: Machine Learning Models Results on Imbalanced Class Distribution

Model	Accuracy	Class	Precision	Recall	F1 Score
RF	0.985	Anomaly	0.98	0.99	0.99
		Normal	0.99	0.97	0.98
		macro avg	0.99	0.98	0.98
ADA	0.972	Anomaly	0.97	0.99	0.98
		Normal	0.98	0.94	0.96
		macro avg	0.97	0.97	0.97
ETC	0.983	Anomaly	0.98	0.99	0.99
		Normal	0.99	0.97	0.98
		macro avg	0.98	0.98	0.98
LR	0.885	Anomaly	0.89	0.94	0.91
		Normal	0.87	0.79	0.83
		macro avg	0.88	0.86	0.87

TABLE V: Machine Learning Models Results using Novel S-DATE Balanced Class Distribution

Model	Accuracy	Class	Precision	Recall	F1 Score
RF	0.991	Anomaly	0.99	0.99	0.99
		Normal	0.99	0.99	0.99
		macro avg	0.99	0.99	0.99
ADA	0.971	Anomaly	0.96	0.98	0.97
		Normal	0.98	0.96	0.97
		macro avg	0.97	0.97	0.97
ETC	0.989	Anomaly	0.99	0.99	0.99
		Normal	0.99	0.99	0.99
		macro avg	0.99	0.99	0.99
LR	0.870	Anomaly	0.85	0.89	0.87
		Normal	0.89	0.85	0.86
		macro avg	0.87	0.87	0.87

In comparison with S-DATE performance, we also evaluate state-of-the-art data balancing approaches. We deploy SMOTE, ADASYN, and CTGAN techniques which are very well-known for data generation for the minority class. RF is significant with all approaches and achieved 0.990 accuracies with SMOTE and CTGAN techniques, while it shows a 0.988 accuracy score. The performance of machine learning models using the SMOTE, ADASYN, and CTGAN is shown in Tables VI, VII, and VIII, respectively. The RF model with SMOTE and CTGAN showed similar results in terms of accuracy, precision, recall, and F1 score. However, the model with ADASYN-generated samples showed slightly lower performance in terms of accuracy. LR is significant with CTGAN-generated samples even as compared to S-DATE because its generated output matches the distributions of the discrete variables in the training data.

Figure 5 shows the comparison between all approaches. We illustrate only high accuracy in the graph from each approach, and similarly, in Figure 6, we show the confusion matrix of the best performer using each approach. The performance of S-DATE is significant in terms of accuracy score and the number of correct and wrong predictions. Using the S-DATE technique, RF achieved 0.991 accuracies and gave 42901 correct predictions out of 43304 test predictions. RF only

TABLE VI: Machine Learning Models Results using SMOTE Balanced Class Distribution

Model	Accuracy	Class	Precision	Recall	F1 Score
RF	0.990	Anomaly	0.99	0.99	0.99
		Normal	0.99	0.99	0.99
		macro avg	0.99	0.99	0.99
ADA	0.966	Anomaly	0.96	0.98	0.97
		Normal	0.98	0.96	0.97
		macro avg	0.97	0.97	0.97
ETC	0.989	Anomaly	0.99	0.99	0.99
		Normal	0.99	0.99	0.99
		macro avg	0.99	0.99	0.99
LR	0.866	Anomaly	0.85	0.89	0.87
		Normal	0.88	0.84	0.86
		macro avg	0.87	0.87	0.87

TABLE VII: Machine Learning Models Results using ADASYN Balanced Class Distribution

Model	Accuracy	Class	Precision	Recall	F1 Score
RF	0.988	Anomaly	1.00	0.98	0.99
		Normal	0.98	1.00	0.99
		macro avg	0.99	0.99	0.99
ADA	0.938	Anomaly	0.94	0.93	0.94
		Normal	0.93	0.94	0.94
		macro avg	0.94	0.94	0.94
ETC	0.987	Anomaly	1.00	0.98	0.99
		Normal	0.98	1.00	0.99
		macro avg	0.99	0.99	0.99
LR	0.814	Anomaly	0.80	0.80	0.80
		Normal	0.80	0.80	0.80
		macro avg	0.80	0.80	0.80

TABLE VIII: Machine Learning Models Results using CTGAN Balanced Class Distribution

Model	Accuracy	Class	Precision	Recall	F1 Score
RF	0.990	Anomaly	0.99	0.99	0.99
		Normal	0.99	0.99	0.99
		macro avg	0.99	0.99	0.99
ADA	0.977	Anomaly	0.97	0.99	0.98
		Normal	0.99	0.97	0.98
		macro avg	0.98	0.98	0.98
ETC	0.975	Anomaly	0.96	1.00	0.98
		Normal	1.00	0.96	0.98
		macro avg	0.98	0.98	0.98
LR	0.900	Anomaly	0.89	0.92	0.90
		Normal	0.91	0.88	0.90
		macro avg	0.90	0.90	0.90

predicted the 403 test example as wrong. Similarly, SMOTE RF gives 417 wrong predictions out of 43304 test predictions, CTGAN gives 423 wrong predictions, and ADASYN gives 484 wrong predictions. On imbalanced data minimum, wrong predictions are by RF which are 493 out of 33130 total test predictions.

B. Results for M-En using only Training Data Re-sampling Approach

Several research suggests data balancing after data splitting and only on the training set because data balancing before splitting can cause for same data leakage into the test set, which is present in the training set [35]. We also did data re-sampling only on the training set after data splitting to show the significance of the proposed S-DATE. Results of the best

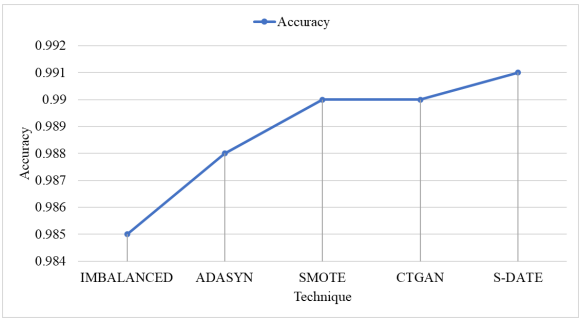
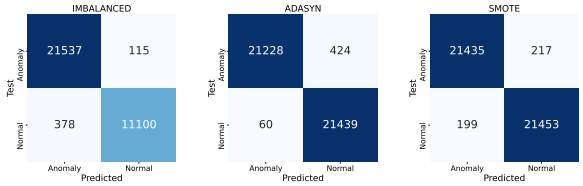
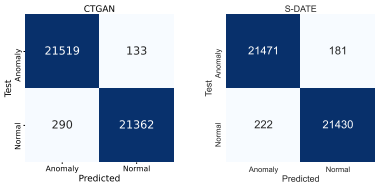


Fig. 5: Comparison of accuracy scores using each data sampling Technique



(a) (b) (c)



(d) (e)

Fig. 6: Confusion matrices using each data sampling Technique

performer with all resampling techniques are shown in Table IX, and results show that S-DATE showing performed significantly with a 0.988 accuracy score. The wrong prediction ratio with S-DATE is the lowest as compared to all approaches, with 381 out of 33130. In this scenario also, RF with S-DATE performs equally well for both classes, with 181 wrong predictions for the Anomaly class and 200 for the Normal class.

TABLE IX: Results when applying data balancing technique after data splitting

Approach	Model	Accuracy	Precision	Recall	F1 Score	CP	WP
S-DATE	RF	0.988	0.99	0.99	0.99	32749	381
SMOTE	RF	0.987	0.99	0.99	0.99	30723	407
ADASYN	RF	0.984	0.98	0.98	0.98	32621	509
CTGAN	RF	0.987	0.99	0.99	0.99	32725	405

C. S-DATE Results for Benchmark Dataset

To show the significance of the proposed S-DATE approach, we deployed it on a benchmark dataset, CSE-CIC-IDS2018. Table X shows the experimental results for the CSE-CIC-IDS2018 dataset [28], which we acquired from Kaggle. The dataset consists of many CSV files, but we

selected only one file, '02-28-2018.csv', for experiments. This file consists of 613,104 samples, and we used 25% of the dataset. The data samples are categorized into two classes: 'Benign' with 134,801 samples, and 'Infiltration (Malicious)' with 17,124 samples. The dataset is highly imbalanced, which leads the model to perform poorly for the minority class, 'Malicious.' The performance of machine learning models with each technique is compared in Table X. According to the results, machine learning models show poor performance in terms of accuracy and F1 score because of training on a highly imbalanced dataset. After deploying data balancing techniques, the performance of models improved in terms of all evaluation parameters. S-DATE, SMOTE, and ADASYN perform approximately equally well, but RF with S-DATE achieves a significant 0.954 accuracy score, followed by RF with ADASYN, which is 0.951. Overall, the performance of S-DATE is a little significant. The performance of CTGAN is not good on the CSE-CIC-IDS2018 dataset as it generates most of the null values in the augmented dataset. So, when we remove null values, the model's performance is poor.

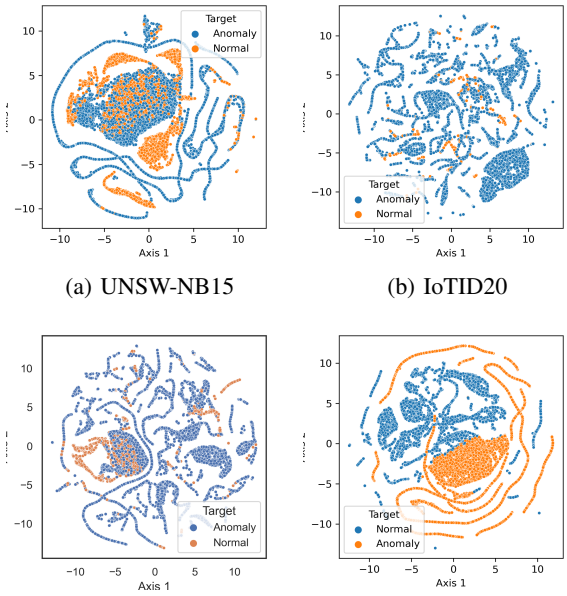
TABLE X: Results on CSE-CIC-IDS2018 Dataset

Approach	Model	Accuracy	Precision	Recall	F1 Score
S-DATE	RF	0.954	0.95	0.95	0.95
	ADA	0.88	0.88	0.88	0.88
	ETC	0.94	0.94	0.94	0.94
	LR	0.57	0.57	0.57	0.57
SMOTE	RF	0.942	0.94	0.94	0.94
	ADA	0.88	0.89	0.88	0.88
	ETC	0.93	0.93	0.93	0.93
	LR	0.55	0.55	0.55	0.55
ADASYN	RF	0.951	0.95	0.95	0.95
	ADA	0.890	0.89	0.89	0.89
	ETC	0.91	0.91	0.91	0.91
	LR	0.55	0.55	0.55	0.54
CTGAN	RF	0.919	0.83	0.72	0.76
	ADA	0.91	0.86	0.65	0.70
	ETC	0.88	0.69	0.59	0.61
	LR	0.989	0.44	0.50	0.47
IMBALANCED	RF	0.919	0.83	0.72	0.76
	ADA	0.91	0.86	0.65	0.70
	ETC	0.88	0.69	0.59	0.61
	LR	0.89	0.44	0.50	0.47

D. Discussion

Network security using artificial intelligence is critical thing because evolving modern network architecture increases the threat of breaching networks' security systems. It is needed for a time; AI-based systems should be strong enough to deal with different kinds of attacks. This study performed experiments on M-En and proposed an approach that can detect malicious traffic either for IoT devices or Tr. IP-based devices. We generate an M-En dataset using the benchmark and very well know datasets IoTID-20 and UNSW-NB15. These datasets are used by researchers to train AI-based network security systems for IoT and Tr. IP-based networks but individually. So to securely analyze both kinds of systems, IoT and Tr. IP-based traffic, two approaches used by the researchers. We overcame this problem and, after generating the M-En dataset, proposed a significant approach that gives a high accuracy score of 0.991 for malicious traffic detection.

Artificial intelligence is crucial for network security due to the increased risk of security breaches in modern network architecture. It is necessary for AI-based systems to be strong enough to defend against various types of attacks. This study conducted experiments on M-En and developed an approach capable of identifying malicious traffic in both IoT and Tr. IP-based devices. This research created the M-En dataset by combining the well-known IoTID-20 and UNSW-NB15 datasets, which are often used to train AI-based network security systems for IoT and Tr. IP-based networks separately. The used datasets are highly imbalanced, which leads models toward overfitting. As shown in Figures 7a and 7b, which show both UNSW-NB15 and IoTID-20 feature spaces, the Anomaly class totally dominates the Normal class. When we generate the M-En dataset, there is also the domination of the Anomaly class as shown in Figure 7c. To address the challenge of analyzing both types of traffic securely, the research proposed a novel approach that achieved a high accuracy score of 0.991 for malicious traffic detection. Figure 7d show the significance of S-DATE for M-En data in which class distribution is equal and linearly separable. S-DATE improves the model's performance by generating very correlated data to the original data and giving equal training to models for both classes.



(a) UNSW-NB15 (b) IoTID20

(c) Imbalanced M-En (d) Balanced M-En

Fig. 7: Feature space for each dataset

E. Comparison with other studies

To show the significance of our proposed approach, we perform a comparison with other studies in the network security domain. There is only one study on M-En [12], so to increase the comparison analysis, we implement other studies approaches on our M-En dataset. To perform a fair comparison, we implement the previous studies' approaches in the same environment which we used for our experiments.

We included recent studies in the comparison analysis, such as the study [12] used an ensemble learning approach for M-En traffic which is a combination of RF, ETC, and GBM. The study [36], deployed used SMOTE technique to overcome the data imbalance problem and then used XGBoost to detect the malicious traffic. The study [37] used RF-Recursive Feature Elimination (RF-RFE) technique with LightGBM for network intrusion detection. The study [2] used a machine learning approach for malicious traffic detection with hybrid feature selection. They used principal comment analysis (PCA) and singular vector decomposition (SVD) features with RF to achieve their best results. Similarly, another study [38] deploys a deep learning approach for malicious traffic detection. They used a recurrent neural network architecture to achieve high accuracy, as shown in Table XI. The results show that our proposed approach is significant in comparison with all approaches.

TABLE XI: Comparison with previous studies

Ref	Year	Approach	Accuracy	Precision	Recall	F1 score
[12]	2021	Ensemble Model	0.985	0.994	0.980	0.986
[36]	2023	SMOTE + XGBoost	0.987	0.987	0.988	0.987
[37]	2023	RF-RFE +LightGBM	0.984	0.989	0.986	0.988
[2]	2022	PCA+SVF+RF	0.980	0.991	0.979	0.985
Our	**2023**	RF with S-DATE	**0.991**	**0.991**	**0.990**	**0.991**

F. Statistical Significant Test

To show the significance of the proposed S-DATE technique we perform a statistical T-test. We compared the results of S-DATE with other techniques and find that is there any statistically significant difference between S-DATE and other technique results. The T-test compares two results and accepts or rejects the null hypothesis (N_h). This N_h can be defined as:

- Accept N_h: T-test accept N_h when two compared results are not statistically significant. When T-test accepts N_h and then it rejects the alternative hypothesis A_h.
- Reject N_h: T-test reject N_h when two compared results are statistically significant. When T-test rejects N_h and then it accepts the A_h.

Table XII shows the results for T-test which deploy on M-En results. We deploy the T-test with a 0.5 value of alpha in all cases and measure in terms of t statistic (T), degree of freedom (DF), critical value (CV), and p-value (P). Results show that the S-DATE approach is statistically significant in comparison with all other approaches.

TABLE XII: Statistical T-test results.

Case	T	DF	CV	P	N_h
S-DATE Vs. IMBALANCED	-1.446	6	0.000	0.198	Reject
S-DATE Vs. ADASYN	-0.786	6	0.000	0.462	Reject
S-DATE Vs. SMOTE	-2.611	6	0.000	0.040	Reject
S-DATE Vs. CTGAN	-0.837	6	0.000	0.435	Reject

VII. Conclusion

The objective of this research was to develop a system for detecting malicious traffic in M-En networks using machine learning techniques. The proposed approach involves a novel data augmentation technique called S-DATE and a hybrid dataset called M-En. Our study identified a significant security issue in M-En networks, which can be addressed using our proposed solution, achieving an accuracy score of 0.991. The RF model with the S-DATE technique outperformed state-of-the-art approaches, such as CTGAN, ADSYN, and SMOTE, for detecting malicious traffic in M-En networks. Furthermore, we found that IoT-based and Tr. IP-based traffic can coexist in the same network, such as smart home, and our proposed approach can effectively handle such networks when imbalanced datasets are available for model training. We concluded that our S-DATE approach is significant because it searches the neighboring space after feature reduction, enabling us to work on important data with optimized performance. To conclude, we have found that our approach is statistically significant when compared to other existing approaches.

Limitations and Future Work: The M-En dataset is based on an existing dataset and may not reflect actual traffic in hybrid networks. Our S-DATE approach was only tested on network security datasets and may behave differently on other types of datasets. Additionally, this study only considered IoT and Tr. IP-based traffic and did not examine SDN-based or other architectures. To overcome these limitations, we will generate real M-En traffic by creating a network and explore more advanced learning models like neural networks in future work to improve the proposed approach. Furthermore, we will deploy the proposed S-DATE technique on a more complex dataset to validate its significance and effectiveness.

Dataset Availability

The M-En dataset, generated as part of this study, is accessible to the public through the following link: https://www.kaggle.com/datasets/furqanrustam118/m-en-dataset.

References

[1] M. Gopinath and S. C. Sethuraman, "A comprehensive survey on deep learning based malware detection techniques," *Computer Science Review*, vol. 47, p. 100529, 2023.

[2] F. Rustam, M. F. Mushtaq, A. Hamza, M. S. Farooq, A. D. Jurcut, and I. Ashraf, "Denial of service attack classification using machine learning with multi-features," *Electronics*, vol. 11, no. 22, 2022.

[3] N. Y. Conteh and P. J. Schmick, "Cybersecurity: risks, vulnerabilities and countermeasures to prevent social engineering attacks," *International Journal of Advanced Computer Research*, vol. 6, no. 23, p. 31, 2016.

[4] D. Resul and M. Z. Gündüz, "Analysis of cyber-attacks in iot-based critical infrastructures," *International Journal of Information Security Science*, vol. 8, no. 4, pp. 122–133, 2020.

[5] I. C. Eian, L. K. Yong, M. Y. X. Li, Y. H. Qi, and Z. Fatima, "Cyber attacks in the era of covid-19 and possible solution domains," 2020.

[6] I. Palo Alto Networks, "2022 attack surface threat report." https://start.paloaltonetworks.com/2022-asm-threat-report. acceesed on 03 April 2023.

[7] Integrity360, "2022 in 22 cyber security statistics." https://insights.integrity360.com/2022-in-22-cyber-security-statistics. acceesed on 03 April 2023.

[8] S. R. Masadeh, A. Azzazi, B. A. Alqaralleh, and A. M. Al Sbou, "A novel paradigm in authentication system using swifi encryption/decryption approach," *International Journal of Network Security & Its Applications*, vol. 6, no. 1, p. 17, 2014.

[9] D. Anderson and N. Kipp, "Implementing firewalls for modern substation cybersecurity," in *proceedings of the 12th Annual Western Power Delivery Automation Conference, Spokane, WA*, 2010.

[10] A. Alotaibi and M. A. Rassam, "Adversarial machine learning attacks against intrusion detection systems: A survey on strategies and defense," *Future Internet*, vol. 15, no. 2, p. 62, 2023.

[11] K. Lin, X. Xu, and F. Xiao, "Mffusion: A multi-level features fusion model for malicious traffic detection based on deep learning," *Computer Networks*, vol. 202, p. 108658, 2022.

[12] P. L. Indrasiri, E. Lee, V. Rupapara, F. Rustam, and I. Ashraf, "Malicious traffic detection in iot and local networks using stacked ensemble classifier," *Computers, Materials and Continua*, vol. 71, no. 1, pp. 489–515, 2022.

[13] C. Davis, "Home network segmentation: A must in the iot era." https://www.ckd3.com/blog/2018/10/15/home-network-segmentation-a-must-in-the-iot-era. acceesed on 03 April 2023.

[14] B. Bowen, A. Chennamaneni, A. Goulart, and D. Lin, "Blocnet: a hybrid, dataset-independent intrusion detection system using deep learning," *International Journal of Information Security*, pp. 1–25, 2023.

[15] K. Dhanya, S. Vajipayajula, K. Srinivasan, A. Tibrewal, T. S. Kumar, and T. G. Kumar, "Detection of network attacks using machine learning and deep learning models," *Procedia Computer Science*, vol. 218, pp. 57–66, 2023. International Conference on Machine Learning and Data Engineering.

[16] Y. Feng and C. Wang, "Network anomaly early warning through generalized network temperature and deep learning," *Journal of Network and Systems Management*, vol. 31, no. 2, pp. 1–34, 2023.

[17] S. Srinivasan and P. Deepalakshmi, "An innovative malware detection methodology employing the amalgamation of stacked bilstm and cnn+lstm-based classification networks with the assistance of mayfly metaheuristic optimization algorithm in cyber-attack," *Concurrency and Computation: Practice and Experience*, vol. 35, no. 10, p. e7679, 2023.

[18] J. Wang, M. Liu, X. Yin, Y. Zhao, and S. Liu, "Semi-supervised malicious traffic detection with improved wasserstein generative adversarial network with gradient penalty," in *2022 IEEE 6th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, pp. 1916–1922, 2022.

[19] F. Alasmary, S. Alraddadi, S. Al-Ahmadi, and J. Al-Muhtadi, "Shieldrnn: A distributed flow-based ddos detection solution for iot using sequence majority voting," *IEEE Access*, vol. 10, pp. 88263–88275, 2022.

[20] M. Sarhan, S. Layeghy, N. Moustafa, M. Gallagher, and M. Portmann, "Feature extraction for machine learning-based intrusion detection in iot networks," *Digital Communications and Networks*, 2022.

[21] R. Kale, Z. Lu, K. W. Fok, and V. L. L. Thing, "A hybrid deep learning anomaly detection framework for intrusion detection," in *2022 IEEE 8th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing, (HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS)*, pp. 137–142, 2022.

[22] P. Jayalaxmi, G. Kumar, R. Saha, M. Conti, T. hoon Kim, and R. Thomas, "Debot: A deep learning-based model for bot detection in industrial internet-of-things," *Computers and Electrical Engineering*, vol. 102, p. 108214, 2022.

[23] Q. Abu Al-Haija, M. Krichen, and W. Abu Elhaija, "Machine-learning-based darknet traffic detection system for iot applications," *Electronics*, vol. 11, no. 4, 2022.

[24] K. A. Da Costa, J. P. Papa, C. O. Lisboa, R. Munoz, and V. H. C. de Albuquerque, "Internet of things: A survey on machine learning-based intrusion detection approaches," *Computer Networks*, vol. 151, pp. 147–157, 2019.

[25] F. Rustam, A. Raza, I. Ashraf, and A. D. Jurcut, "Deep ensemble-based efficient framework for network attack detection," in *2023 21st Mediterranean Communication and Computer Networking Conference (MedComNet)*, 2023. Accepted for publication.

[26] N. Moustafa and J. Slay, "Unsw-nb15: a comprehensive data set for network intrusion detection systems (unsw-nb15 network data set)," in *2015 military communications and information systems conference (MilCIS)*, pp. 1–6, IEEE, 2015.

[27] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the kdd cup 99 data set," in *2009 IEEE symposium on computational intelligence for security and defense applications*, pp. 1–6, Ieee, 2009.

[28] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in *International Conference on Information Systems Security and Privacy*, 2018.

[29] I. Ullah and Q. H. Mahmoud, "A scheme for generating a dataset for anomalous activity detection in iot networks," in *Canadian Conference on Artificial Intelligence*, pp. 508–520, Springer, 2020.

[30] S. Dadkhah, H. Mahdikhani, P. K. Danso, A. Zohourian, K. A. Truong, and A. A. Ghorbani, "Towards the development of a realistic multidimensional iot profiling dataset," in *2022 19th Annual International Conference on Privacy, Security & Trust (PST)*, pp. 1–11, 2022.

[31] K. Taunk, S. De, S. Verma, and A. Swetapadma, "A brief review of nearest neighbor algorithm for learning and classification," in *2019 International Conference on Intelligent Computing and Control Systems (ICCS)*, pp. 1255–1260, IEEE, 2019.

[32] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.

[33] H. He, Y. Bai, E. A. Garcia, and S. Li, "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, 2008.

[34] E. Lee, F. Rustam, W. Aljedaani, A. Ishaq, V. Rupapara, and I. Ashraf, "Predicting pulsars from imbalanced dataset with hybrid resampling approach," *Advances in Astronomy*, vol. 2021, pp. 1–13, 2021.

[35] F. El Barakaz, O. Boutkhoum, M. Hanine, A. El Moutaouakkil, F. Rustam, S. Din, and I. Ashraf, "Optimization of imbalanced and multidimensional learning under bayes minimum risk and savings measure," *Big Data*, vol. 10, no. 5, pp. 425–439, 2022.

[36] M. A. Talukder, K. F. Hasan, M. M. Islam, M. A. Uddin, A. Akhter, M. A. Yousuf, F. Alharbi, and M. A. Moni, "A dependable hybrid machine learning model for network intrusion detection," *Journal of Information Security and Applications*, vol. 72, p. 103405, 2023.

[37] G. Logeswari, S. Bose, and T. Anitha, "An intrusion detection system for sdn using machine learning.," *Intelligent Automation & Soft Computing*, vol. 35, no. 1, 2023.

[38] C. Yin, Y. Zhu, J. Fei, and X. He, "A deep learning approach for intrusion detection using recurrent neural networks," *IEEE Access*, vol. 5, pp. 21954–21961, 2017.