RTL-DL: A HYBRID DEEP LEARNING FRAMEWORK FOR DDOS ATTACK DETECTION IN A BIG DATA ENVIRONMENT

Hassan A Afolabi and Abdurazzag A Aburas

[1]School of Electrical, Electronic and Computer Engineering,
University of Kwazulu-Natal South Africa

ABSTRACT

A distributed denial of service (DDoS) attack is one of the most common cyber threats to the Internet of Things (IoT). Several deep learning (DL) techniques have been utilized in intrusion detection systems to prevent DDoS attacks. However, their performance is greatly affected by a large class imbalance nature of the training datasets as well as the presence of redundant and irrelevant features in them. This study proposes RTL-DL, a new framework for an effective intrusion detection model based on the random oversampling technique and the Tomek-Links sampling technique (RTL), to minimize the effects of data imbalance in the CICIDS2017 dataset used to evaluate the proposed model. This study achieved 98.3% accuracy, 98.8% precision, 98.3% recall, 97.8% f-score, and 4.6% hamming loss. In comparison to current approaches, the suggested model has demonstrated promising results in identifying network threats in imbalanced data sets.

KEYWORDS

Random oversampling, Tomek Links, Internet of things, Imbalanced data, Distributed denial of service, Intrusion detection, Deep Learning and Feature extraction.

1. INTRODUCTION

The Internet of Things (IoT) has proven to be an emerging technology, with recent improvements allowing massive amounts of data to be collected from a variety of sources and in a variety of formats. Zettabytes are the units of measurement for the amount of data available today (ZB). One trillion gigabytes (GB) equals one zettabyte (ZB). According to IDC [1], the total amount of digital data worldwide is estimated to reach 175 ZB in 2025. The enormous amount of digital data generated by IoT in recent times is known as big data. IoT implementation has many characteristics that pose a number of security risks.

Cyber security is a major issue because poorly secured IoT nodes can be readily attacked [2]. Currently, a distributed denial of service (DDoS) attack is one of the most prevalent cyberthreats. It is a type of attack in which numerous agents known as zombies send a certain number of messages to the same machine, known as the target system [3]. The target system is unable to provide resources and services to its legitimate users as a result of the attack. DDoS attacks have risen considerably in recent years, interrupting numerous IoT networks and resulting in catastrophic losses [4]. Because there could be thousands of zombies, detecting these attacks becomes extremely difficult. As a result, differentiating legitimate traffic from DDoS attack traffic remains a major issue [5].

An Intrusion Detection System (IDS) is a network security device that detects malicious activity from transmitted packets [6]. IDSs are mainly divided into two categories: misuse detection techniques, which analyze network behavior using signature matching algorithms for known cases of misuse to detect intrusions; This technique is less effective at detecting novel attacks. Anomaly detection is the second category, which is designed to create a "normal" model of a user's activities and classify any variation from that standard as an intrusion. Although both kinds of IDS have benefits and drawbacks, the anomaly detection technique is more effective at detecting novel attacks. Several researchers have successfully deployed classical machine and deep learning algorithms to network intrusion detection systems as artificial intelligence technology advances. However, it is challenging for machine learning- based intrusion detection systems to detect new DDoS attacks. However, deep learning techniques such as Convolutional Neural Network (CNN), Multi-Layer Perceptron (MLP), Recurrent Neural Network (RNN) [7-9] and other related algorithms have a significant improvement in accuracy in comparison to typical machine learning methods. However, they are computationally complex and take a long time to learn. Advances in computer processing power and neural network algorithms have resulted in improved results for deep learning applications on large amounts of network data. Unfortunately, numerous researchers in the past have prioritized developing a model with greater performance regardless of the techniques involved, while paying less attention to the imbalanced nature of existing network intrusion detection datasets.

An existing dataset is said to be imbalanced if the number of normal (benign) traffic samples in a training set is significantly greater than the number of minority attack samples. Any model trained on such a dataset will produce biased results. Because the primary goal of an IDS is to detect abnormal traffic, concentrating on normal (benign) traffic in the majority of sample data will result in a considerable decrease in the detection accuracy of abnormal (attack) traffic in the minority samples [10]. Furthermore, before deploying any detection method, relevant features from a dataset should be selected to improve the accuracy and efficiency of IDS. A feature selection technique is an effective preprocessing method for an IDS [11,12]. It identifies important features and removes those that are irrelevant. In this paper, we present a framework for mitigating DDoS attacks on an IoT platform while considering issues such as class imbalance, computational complexity, and longer training time. The following are the main contributions of our work:

i. Current state-of-the-art deep learning techniques for detecting DDoS attacks were outlined and categorized by their methods, benefits, and limitations.
ii. Highlighted and troubleshot some of the problems of scattered presence, high class imbalance nature, and irrelevant and redundant features in the CICIDS2017 dataset.
iii. Presented an IDS technique that is based on a DL algorithm for effectively detecting DDoS attacks in a big data environment.
iv. Improve classification performance by using the mutual information gain feature selection technique and RTL (random over-sampling and the Tomek-Links under-sampling) to reduce a large number of irrelevant features in the extracted samples and handle the problems of class imbalance in the experimental dataset, respectively.
v. The proposed model makes a significant contribution to big data research by successfully classifying DDoS attacks on network traffic data and outperforming some state-of-the-art techniques widely used in cybersecurity when compared to it.

The remaining part of the paper is structured as follows: The main focus of Section 2 is a quick summary of related studies. The materials and methods of this work are described in Section 3. The experimental framework was presented in Section 4 and the results were discussed in Section 5. In Section 6, the proposed model was compared to some "state-of-the-art" models, which were talked about in Section 2. In Section 7, conclusions were made.

2. RELATED WORKS

DDoS attacks on IoT networks have notably increased in popularity. The Mirai botnet's October 2016 attack on Dyn Server, a company that manages DNS infrastructure, is an example. This attack had a massive economic impact on major digital companies like PayPal and Amazon. In April 2017, Trend Micro cybersecurity studies found a DDoS attack that used the Persirai botnet and attacked more than 100 IP camera models. Several studies have been conducted to mitigate DDoS attacks in the IoT. The related studies discussed in this section are categorized into three, namely classification algorithms, imbalanced datasets, and feature selection.

2.1. Classification Algorithms

The authors in [13] proposed a method for identifying DDoS attacks in LoT networks based on a deep learning approach. The ISCX, NSL-KDD, and KDDCUP99 datasets were used to test their model. For the detection of unknown DoS/DDoS attacks, DNN and LSTM models were presented by Sabeel et al. [14]. Their models were trained on preprocessed DoS and DDoS samples from the CICIDS2017 dataset and then evaluated on the synthetic ANTS2019 dataset for accuracy. Their models were then retrained on the merged synthetic dataset with the CICIDS2017 dataset and the detection performance against newly synthesized unknown attacks was evaluated. The accuracy of the DNN and LSTM models in the second phase of the experiment was 98.72 percent and 96.16 percent, respectively.

Bandwidth and connection flooding DDoS attacks were the main focus of Virupakshar et al [15]. The authors employed DT, DNN, KNN, and NB algorithms to detect DDoS attacks in an OpenStack-based cloud. The authors compared these classifiers on a dynamically generated dataset and selected the DNN model because it has a higher accuracy and precision value. For cloud datasets, the DNN classifier outperformed the DT, KNN, and NB classifiers in terms of precision and accuracy. However, when compared to other algorithms, the DNN algorithm has a lower precision value for the KDDCUP99 dataset. The use of an old KDDCUP99 dataset, as well as the lack of information about the LAN and cloud datasets, is a major limitation of their study. A feed-forward back-propagation-based DNN architecture called DeepDetect was presented by Asad et al. [16]. DeepDetect was evaluated on the CICIDS2017 dataset. When DeepDetect was compared to RF and DeepGFL, it outperformed the other approaches with an F1-score of 0.99 and an AUC value close to one. This approach was only evaluated for application-layer DDoS attacks.

An FC feed-forward deep neural network model was proposed by Muraleedharan and Janet [17]. The model is based on flow data for detecting slow DoS attacks on HTTP. Only the DoS samples in the CICIDS2017 dataset was used to evaluate the model, and they achieved an accuracy of 99.61%. To detect data flooding attacks in MANETs, Sbai and El Boukhari [18] presented a DNN-based DL model. Their model was trained and evaluated only on the data flooding attack samples from the CICDDoS2019 dataset. The proposed model achieved promising results such as 0.99 for precision, F1-score, and accuracy, respectively, and a recall of 1. In Amaizu et al. [19], they presented a deep learning framework for detecting DDoS attacks in 5G and B5G environments. Their model was developed by joining two different DNN models that were integrated with a feature selection algorithm. Their model was evaluated on the CICDDoS2019 dataset, and it achieved 99.66% accuracy with a loss of 0.011. Furthermore, their framework was compared to the existing CNN ensemble, DeepDefense, KNN, and SVM techniques. Their framework outperformed all except the CNN ensemble in terms of precision and recall. The complex structure of the proposed model is a limitation of their study as this may prolong detection time and affect the model’s performance in a real-time scenario.

Cil et al [20] presented a DL model with feature extraction and classification module. The CICDoS2019 dataset, which was split into two, namely Dataset 1 and Dataset 2 for binary and multi classification, respectively, were used to evaluate their model. Their model achieved about 100% and 95% accuracy rates for detecting DDoS attacks on dataset1 and dataset2, respectively.

Authors in [21] presented a stacking technique to detect network anomalies. They apply KNN, AdaBoost and Random Forests and used the Logistic Regression algorithm to automatically search for better parameters for the Stacking model. Their model was evaluated on NSL- KDD 2019 dataset.

2.2. Imbalanced Datasets

A dataset is said to be imbalanced. If the amount of benign traffic in a training dataset is considerably higher than the proportion of attack traffic, any model trained on an unbalanced dataset will be skewed towards the majority. There are three main methods to address the problem of class imbalance in datasets, namely: cost-sensitive technique, classifier-specific solution, and resampling technology. Our focus will be on resampling technology, which is divided into oversampling and under-sampling techniques. Minority samples are generated using oversampling methods to achieve a nearly balanced number of samples in each category, whereas majority of samples are purged using under-sampling methods to achieve a rough balance in each. Several studies have demonstrated that oversampling techniques can achieve better results on data imbalance problems [22]. Due to this, oversampling techniques are mostly used in the field of intrusion detection systems.

Authors in [23] established that the SMOTE algorithm may effectively increase the accuracy of the system through simulation experiments on KDD CUP99 datasets. Khoshgoftaar [24] analyzed deep learning techniques using unbalanced classroom data. Karatas et al. [25] used the SMOTE oversampling method and six traditional machine learning algorithms for classification. The evaluation was done on the CICIDS2018 dataset. Their model outperforms existing models, according to experimental results. Zhang et al. [26] proposed a data processing approach that combines SMOTE oversampling and clustering undersampling based on a Gaussian mixture model. When compared to other traditional oversampling algorithms on the UNSW-NB15 and CICIDS2017 datasets, the results outperform other existing traditional algorithms. Even though there have been a number of studies on class imbalance problems in benchmark datasets, most of them have focused on the SMOTE algorithm, which has its own problems and limits.

2.3. Feature Selection

The process of determining if a feature is relevant or not for a particular classification problem is known as feature selection. It is a necessary step in the data processing process prior to employing a machine learning algorithm. Furthermore, feature selection [27, 28] can be used as a preprocessing step in machine learning algorithms to reduce computational complexity. It seeks to eliminate superfluous features whilst also conserving or even enhancing the IDS's performance. Hota and Shrivas [11] presented a model that used various feature selection methods to remove unnecessary features in order to develop a more robust and effective classifier. The study demonstrated that when only 17 characteristics from the NSL-KDD dataset are used, C4.5 with information gain achieves the highest accuracy. Abdullah et al. [29] also presented an IDS framework based on feature selection within the NSL-KDD dataset. An information gain (IG) filter was used to merge several sub-sets of the input dataset into their framework.

2.4. Research Gaps of Related Works

Some of the studies discussed in the related studies above focused on DDoS attack detection based on deep learning approaches, while others focused on class imbalance problems and the selection of relevant features in benchmark datasets. Effective classification, data instability, and the existence of redundant and irrelevant features in the dataset were issues that were only briefly addressed in a small number of studies. Following a quick evaluation of related works summarized in the previous subsection, the following research gaps were found, as indicated in table 1.

Table 1. Summary of research gaps

Research Gapsin Related Works	References
Lack of comprehensive dataset	[14-15],[17],[19-20]
Availability of skewed samples of DDoS attacks instances as compared to legitimate events in the existing datasets	[14-15],[17].
Evaluation using offline dataset	[17-20]
Real- defensive system deployment is not timeautom ated.	

Since the unavailability of a large dataset, the problems of class imbalance, and the deployment of these models in real-world networks are unanswered questions, the study was conducted in a big data scenario to solve large data unavailability. The proposed model was integrated with random oversampling and the TomekLinks under-sampling (RTL) technique to handle class imbalance problems and the information gain technique to extract the most important attributes. The proposed model is computationally efficient and able to detect DDoS attacks in a high-speed automated manner because processing time was reduced considerably with the application of the RTL algorithm.

3. MATERIALS AND METHODS

The materials and techniques used for this study are described in this section. Firstly, the dataset is described, followed by a discussion of the preprocessing methods utilized. Theoretical principles for deep learning approaches are also explained. Benign and DDoS attack samples are extracted from the CICIDS2017 dataset. The extracted samples are preprocessed and normalized in the range [0,1]. The normalized data is inputted into the RTL module to generate synthetic samples of the minority class. The most important features are then selected using a feature selection method called information gain [30]. The selected features are used to train a deep learning algorithm model. Simulation experiments were performed with k-fold cross-validation. A DNN based on back propagation with a Relu activation function is used in the proposed DL model. By adopting the sigmoid function and binary cross-entropy, the proposed model identifies the attack as normal or abnormal.

3.1. Dataset Description

We evaluated our model on the CICIDS2017 dataset, a modern labeled dataset for network intrusion detection that is similar to real-world network data [31]. created by the Canadian Institute for Cybersecurity. The dataset contains 2.83 million records, 85 network flow features, a label attribute, and 225,742 instances with both attack and normal data. In the dataset, there are seven common updated families of attacks, which are shown in table 2 below.

Table 2. Class Distribution of CICIDS2017

CLASS	No of Samples
BENIGN	2,273,097
DDoS	128,027
DoS (hulk, goldeneye,slowloris,slow httptest)	252,661
Portscan	158,930
Patator(FTP,SSH)	13,835
WebAttack	2,180
Bot	1,966
Infiltration	36
Heartbleed	11
Total	**2,830,743**

Since DDoS is a kind of DoS attack and since the focus of our work is on the classification of DDoS attacks, we extracted the DoS and DDoS attack samples in the dataset and modified them in terms of both attack and normal traffic, as illustrated in figure 1 below.

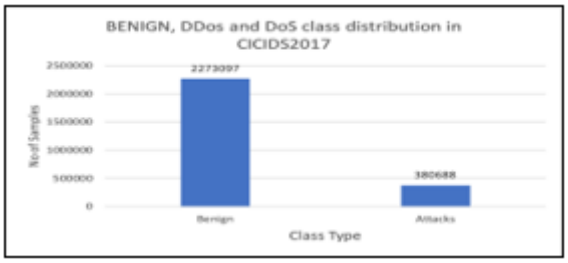

Figure 1. Extracted Samples in Dataset

3.2. Dataset Preprocessing.

Data preprocessing is a step in which raw data is prepared for additional processing. In general, real- world data is inadequate, inconsistent, and riddled with errors. Data preprocessing is a technique for dealing with such issues.

3.2.1. Normalization.

Data normalization is an essential step when implementing machine learning. The attribute data is sized to fall within a small range that is defined, such as -1.0 to 1.0 or 0 to 1.0 in this form of data transformation. In this study, the min–max scaler shown in equation 1 is the technique adopted for normalization to reduce the differences in different dimensions. Min-max normalization scales the data to the interval [0, 1] through a linear transformation.

$$\widetilde{X_{jk}} = \frac{X_{jk} - minX_j}{max(X_j) - min(X_j)} \quad (1)$$

Where $max(X_j)$ and $min(X_j)$ denotes the jth feature's maximum and minimum values, respectively, and X_j and $\widetilde{X_{jk}}$ denotes the normalized feature value in the range [0, 1]. The normalization process prevented certain featureswith large numerical values from affecting the algorithm's outcome and limiting model performance.

3.2.2. Handling Class Imbalance.

A training dataset is said to be imbalanced if the instances of benign traffic are significantly greater than the attack traffic. During the testing and validation phases, any model trained on an imbalanced dataset will be biased toward the majority [32-34]. CICIDS2017 is a dataset with imbalanced classes, as illustrated in Figure 2.

Although imbalanced problems are frequently addressed in traditional machine learning models, deep learning approaches do not take them into account sufficiently. Data used for research purposes would be considerably reduced if data balancing were exclusively accomplished with under-sampling techniques, which would eliminate useful normal network traffic. The use of oversampling techniques alone results in an unnecessary data size increase and noise. In this study, the RTL (ROS + Tomek-Link) technique was used to generate more samples of the minority (attack) class.

3.2.2.1. Random Oversampling

Random oversampling (ROS) is the simplest form of oversampling technique to balance the imbalanced nature of the dataset. It balances out the data by replicating minority class samples. There is no data loss as a result of this.

3.2.2.2. Tomek-Links

TomekLinks is an under-sampling method for imbalanced datasets developed by Tomek in 1976 [35]. Samples on the Tomek links are identified from the dataset in this approach. The sample pairs are data in the data set that are similar but belong to different classes. These data pairs are referred to as Tomek links. The fundamental idea is to differentiate between the minority and majority classes. Let a be a member of one class and b be a member of another, a and b being the closest neighbors, and d (a, b); Assuming that a and b are separated by a distance;

$$T\ (a, b) \text{ is a TomekLink if for any instance i;}$$

$$d\ (a, b) < d\ (a, i)\ or\ d\ (a, b) < d\ (b, i). \tag{2}$$

Figures 2(a) and 2(b) below shows a pie chart of the actual and RTL Generated datasets.

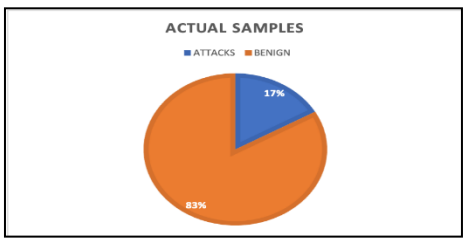

Figure. 2(a). Actual Samples

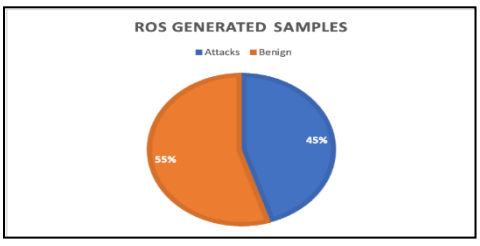

Figure. 2(b). Generated Samples

3.2.3. Data Partitioning.

Since the experimental dataset was not split into training and test sets, we applied K-fold cross-validation. where K denotes five-fold cross-validation. Figure 3 depicts the number of attacks and benign classes in a set of training and testing sets.

Figure 3. Training and Test Set

3.2.4. Feature Selection

In this study, we used an information gain feature selection approach to select important features from the experimental dataset. This technique is a single attribute evaluator in combination with the Ranker search method to score all attributes based on their information gain. The score is determined by how much information about the classes is obtained when a feature is used. A feature can be ignored without affecting the accuracy of a model if it has a very low information gain score [30]. The algorithm below is used to compute and select the best feature subset:

Algorithm 1 : Compute IG and select best feature subset

Input : Training data with *Z* feature numbers

Output : Choose top *n* features as the new set of features

Step 1 : Compute Information Gain:$IG(Y;X) = H(Y) - H(Y|X)$ (3)

Where:$H(Y) = -\sum_{i-1}^{n} P(Y = y_i)\log_2 P(Y = y_i)$ (4)

$H(Y|X) = -\sum_{i-1}^{m} P(X - x_i)H(Y|X = x_i)$. (5)

Step 2 : List features according to their IG score (from highest to lowest)

Step 3 : Compute Total IG = $\sum_{i=1}^{Z} IG(Y;x_i)$. (6)

Step 4 : For each feature x, ascribe weights W (according to its IG with respect to the sum of IG of all features)

$$W(x_i) = \frac{IG(Y;x_i)}{Total\ IG} \quad (7)$$

Step 5 : Set a threshold value *T* and choose the top *n* features (Sum of weights of selected features≥ the threshold value)

Begin from $i = 1(feature\ with\ the\ highest\ IG)$

$$W_n = (W_n + W_{x_i}) \quad (8)$$

If $W_n \geq T$

Move to step 6

Else Increase and repeat previous step 5

Step 6 : Output top *n* features as the new feature set, *n=i*

As illustrated in algorithm 1, Z is the number of input features in the training data setprior to IG, and X is a parameter that indicates individual input features or attributes. ($X_1 \ldots\ldots X_n$) and *Y* is a variable that represents class attributes (DDoS/benign: $Y_1 \ldots\ldots Y_n$). The likelihood that the class attribute *y* occurs is $P\ (Y = y_1)$, the information gain of attribute *X* is IG (*Y;X*), the entropy of *Y* is *H(Y)*, and the average conditional entropy of *Y* is*H(Y|X)*. The threshold value T is dependent on the training dataset used for experimentation, for this research, only 20 attributes were selected when considering threshold values 1.37 (Where n=20, T= 1.37). The new selected features as shown in the table below are then forwarded to the deeplearning model.

Table 3. Description of selected Features

Description	Information	Description	Information
Average	2.01	Init_win_bytes_forward	1.37
Packet	2.00	Total	1.80
Packet	1.86	Max	1.85
Packet	1.86	Init_win_bytes_backward	1.63
Subflow	1.59	Fwd	1.76
Total	1.59	Flow	2.15
Bwd	1.38	Destination	1.38
Bwd	1.51	Flow	2.31
Average	1.50	Flow	1.94
Subflow	1.81	Fwd	2.26

3.3. Deep Learning Model.

Deep learning is a kind of artificial neural network (ANN) that consists of more than one hidden layer [36]. In our study, we constructed a fully connected deep learning model from scratch, namely a deep neural network (DNN). The bedrock of the proposed DNN model is the back-propagation algorithm. Backpropagation begins at the last layer and successively moves back one layer at a time, computing the error for each visited layer. The proposed model has four hidden layers in addition to one input layer and one output layer each. The number of input neurons in the model is the same as the selected features extracted from the experimental dataset in the second component of the overall framework. Therefore, the input training sample with n = 20 inputs is expressed as follows:

$$I=[i_1, i_2 \ldots\ldots i_n] \tag{9}$$

The subsequent layer is the hidden layer h which maps the input I from the input layer with random initialized weights wa and a bias b_j. Thus, hidden layer inputs are expressed as follows:

$$h_j=\sum_a w_a i_a + b_j \tag{10}$$

Where j = [1, 2, ... n_{th}] is the number of hidden nodes in the dnn model. Flowchart of the proposed deep learning model is shown in the figure 4.

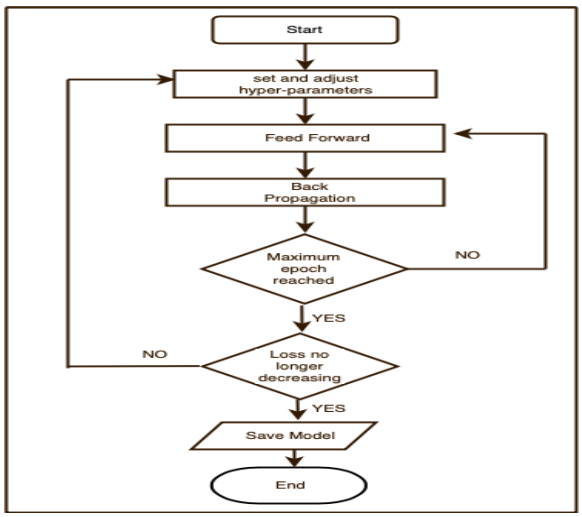

Figure 4. Flowchart of the BPDNN model

Training the proposed model involves four phases, namely weights initialization, feed-forward phase, back propagation, update weights, and bias. The Bernoulli normal distribution is used to initialize the weights, and the training sample is propagated through all four hidden layers during the feed-forward stage to compute the weighted inputs of each layer. Activation function σ is applied to the sum of weighted input signals at each hidden unit and sent to the next layer till it gets to the output unit. During back propagation, the output unit compares its computed activation Yo with its target value T to obtain the error at that unit. The error δo is computed and sent back to all units in the previous layer. Similarly, error δj is computed at each hidden unit hj and sent to the next previous layer. The weight and biases are updated using the error factor δ and the activation function σ till the stop conditions are reached. Details of the hyper-parameters used in training our model are given in the table below:

Table 4. Hyper-parameters description

Descriptions	Values
Objective	Binary classification
Inputvector	20
Hiddenlayers	4
No of Neurons at hidden layer1-4respectively	64,32,16,5
Output	1
Learningrate	0.001
Activation functions	ReLu, Sigmoid
Optimization	Adam optimizer
Loss Function	Binary cross entropy
Momentum	0.9
Weight initializer	Bernoulli normal distribution
Epochs	150

4. FRAMEWORK OF THE PROPOSED MODEL

The RTL-DL model shown in figure 5 is made up of three major components namely the RTL (ROS + Tomek-Link) module, the information gain module, and the deep neural network classifier.

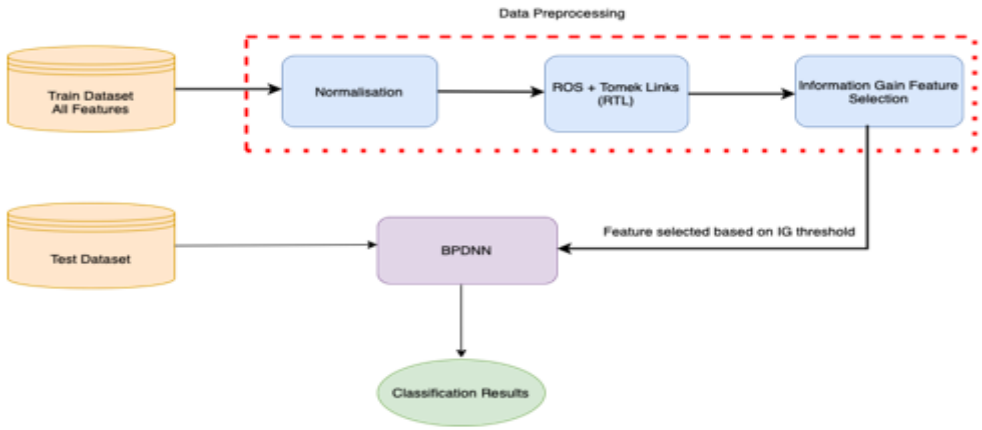

Figure 5. Framework of the Proposed RTL-DL Model

The components outlined above operates sequentially. Minority class samples of the training data are generated to an acceptable level by the RTL component. The relatively balanced generated data by RTL is used as an input for the feature selection component. The IG of all features in the RTL

generated data is computed by the feature selection component using equation 3, as described in algorithm 1. These features are ranked based on their IG values, and the best feature subset is selected prior to applying our BPDNN algorithm.

5. RESULTS AND DISCUSSION

This section discusses the software and hardware requirements used in this experiment. The experiment is based on the RTL-DL, a supervised deep learning algorithm for intrusion detection. The proposed DL model adopts the ReLu activation function for the excitation of nodes at the hidden layers, and a sigmoid activation function is used by nodes at the output layer. The model is evaluated on the benchmark CICIDS2017 dataset. The experiments were performed on a Jupyter notebook version 6.0.3 in the Anaconda environment using Python 3, Keras, and TensorFlow and a Graphics Processing Unit (GPU) installed on a Mac OS X 2.8 GHz Intel Core i7 CPU, 16.00 GB RAM, 2133MHz LPDDR3. The table below shows results obtained during the evaluation of the proposed model on a test dataset.

Table 5. Results Obtained after Testing

Metrics	**Values (%)**
Accuracy	98.3
F-score	97.8
Hamming Loss	4.6
Precision	98.8
Recall	98.3

During intensive experiments to verify the performance of the RTL algorithm for solving class imbalance problems in benchmark cicids2017 datasets, we applied RTL to the extracted samples of the minority class in the original dataset. This increases its number of instances from 360,688 to 1,022,894 in both training and testing sets. Using 79 features in the generated samples, there is an increase of 19.2% in accuracy and 66.3% in fscore, which is one of the most important metrics to use in a class imbalance problem scenario. Also, to verify the performance of feature selection in our experiment, we applied mutual information gain on the RTL-generated samples and obtained an increase of 0.5%, 8.3%, 8.2%, and 5.4% for accuracy, precision, recall, and fscore, respectively, when the top 30 features are selected. Furthermore, there is an increase of 0.1% when the top 20 features are selected as illustrated in table 6.

Table 6. Performance Result of Proposed Approach

	EXPERIMENTALMODEL		**PROPOSED MODELS**	
Metrics	DNN	DNN+RTL	DNN+RTL+IG	DNN+RTL+IG
Feature No	79	79	30	20
Accuracy (%)	78.3	97.5	98.2	98.3
Precision (%)	41.1	90.5	98.8	98.8
Recall (%)	22.1	90.1	98.3	98.3
Fscore (%)	26.1	92.4	97.8	97.8
Processing Time (secs)	580	406	116	103

Additionally, the processing time was also reduced considerably with the application of the RTL algorithm and information gain technique as shown in figure 6 below.

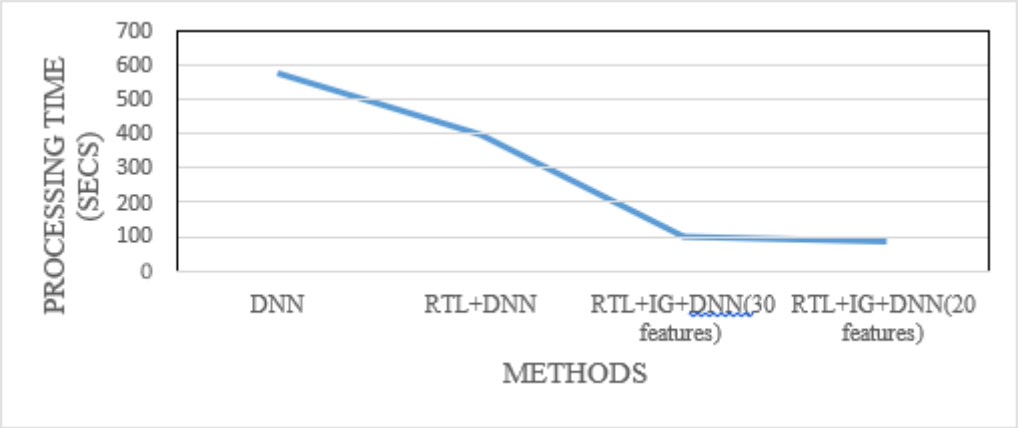

Figure 6. Processing times

6. COMPARATIVE ANALYSIS

The proposed RTL-DL's performance was compared to some state-of-the-art algorithms mentioned in the literature and the results are shown in table 7. The proposed method outperforms the deep neural network (DNN) and random forest (RF) methods in [16] and [22], respectively, in terms of accuracy. The proposed method also outperformed the KNN and LDA methods in [24] and all the methods in [28]. The methods in [17–20] outperformed the proposed method. However, the proposed method included data balancing and feature selection techniques, which were not considered in [17–20] and may affect the performances because of the significant number of irrelevant features and large class imbalance present in the datasets.

Table 7. Brief comparison with other works

Ref/Year	Dataset	Method	Accuracy
[16]/2020	CICIDS2017	DNN	98%
[17]/2020	CICIDS2017	FeedForward DNN	99.61%
[18]/2020	CICDDoS2019	DNN	99.99%
[19]/2021	CICDDoS2019	Composite DNN	99.66%
[1]/2021	CICDDoS2019	DNN	99.99%: Binary 94.57%: Multiclass
[22]/2019	KDDCUP'99	Random Forest+SMOTE	92.57%
[24]/2020	CSE-CIC-IDS2018	(KNN, RF, GB, AB, DT, and LDA) + SMOTE	95.30%, 99.19%, 99.38%,99.32%, 98.56% and83.62%
[28]/2020	NSL-KDD	J48, RF, and PART.	86.0807%, 86.5117%and 86.6606%
Proposed Method	CICIDS2017	DNN +RTL	98.3%

7. CONCLUSIONS

Problems of secure communication have increased exponentially with the recent advancement of IoT networks. The distributed denial of service (DDoS) attack has become one of the most common cyber threats, necessitating the need for effective DDoS prevention and detection. An efficient IDS

to detect distributed denial of service (DDoS) attacks on an IoT platform using a deep neural network based on a back-propagation technique was presented in this paper. The model is based on RTL and Mutual Information Gain techniques to solve class imbalance problems and extract the most relevant features from our experimental dataset, respectively. The further comparison between our experimental models and proposed models proved the importance of oversampling and feature selection techniques in developing state-of-the-art models. This research is only focused on detecting DDoS attacks. Other sorts of cyber-attacks can thus be investigated for future research. Additionally, hyper-parameter tuning so as to decrease the number of epochs required and further decrease the training time will be considered. Finally, the combination of the presented model with other machine learning techniques to be able to detect more novel attacks and evaluate them on other recent benchmark datasets will be considered.

CONFLICTS OF INTEREST

The authors declare no conflict of interest.

LIST OF ABBREVIATIONS

AdaBoost: Adaptive Boosting
BP: Back propagation
CICIDS2017: Canadian Institute of Cybersecurity Intrusion detection dataset
CNN: convolutional neural network
DDoS: Distributed Denial of Service Attacks
DNN: Deep Neural Network
DL: Deep Learning
DT: Decision Tree
FC feed-forward: Fully Co
FC feed-forward: Fully Connected feed-forward
GB: Gigabytes
IDS: Intrusion Detection System
IoT: Internet of Things
IG: Information Gain
KNN: k-nearest neighbors' algorithm
KDDCUP99: Knowledge Discovery and Data Mining
LSTM: Long short-term memory
LDA: Linear Discriminant Analysis
NSL-KDD: Knowledge Discovery and Data Mining
ROS: Random Oversampling
TL: TomekLink
ReLu: Rectified Linear Unit
RNN: Recurrent neural network
SMOTE: Synthetic Minority Oversampling Technique
ZB: Zettabyte

REFERENCES

[1] Reinsel, D., Gantz, J., &Rydning, J. (2017). IDC white paper: Data Age 2025: "The Evolution of Data to Life-Critical".
[2] Chadd, A. (2018). DDoS attacks: past, present and future. Network Security, 2018(7), 13-15. https://doi.org/10.1016/S1353-4858(18)30069-2

[3] Mukkavilli, S.K., Shetty, S. and Hong, L. (2016) Generation of Labelled Datasets to Quantify the Impact of Security Threats to Cloud Data Centers. Journal of Information Security, 7, 172- 184. https://doi.org/10.4236/jis.2016.73013
[4] C. Kolias, G. Kambourakis, A. Stavrou, and J. Voas, 'DDoS in the loT: Mirai and other botnets', Computer, vol. 50, no. 7, pp. 80-84, 2017.
[5] Wagholi, P. (2014). Detection of DDoS Attacks Based on Network Traffic Prediction and Chaos Theory. International Journal of Computer Science and Information Technologies, Vol. 5 (5), 2014, 6502-6505.
[6] Atkinson RC, Bellekens XJ, Hodo E, Hamilton A, Tachtatzis C (2017) Shallow and deep networks intrusion detection system: a taxonomy and survey. CoRR, arXiv preprint arXiv:1701.02145. 2017 Jan 9
[7] Liu Y, Liu S, Zhao X. In: DEStech Transactions on Engineering and Technology Research (ICETA). Intrusion detection algorithm based on convolutional neural network; 2017.
[8] Esmaily J, Moradinezhad R, Ghasemi J. Intrusion detection system based on multi-layer perceptron neural networks and decision tree. In: 2015 7th Conference on Information and Knowledge Technology (IKT). IEEE; 2015. p. 1–5.
[9] Yin C, Zhu Y, Fei J, He X. A deep learning approach for intrusion detection using recurrent neural networks. IEEE Access 2017;5:21954–61.
[10] Thabtah F, Hammoud S, Kamalov F, Gonsalves A. Data imbalance in classification: experimental evaluation. Inf. Sci. 2020;513:429–41.
[11] H. Hota, A.K. Shrivas, Decision Tree Techniques Applied on Nsl-kdd Data and Its Comparison with Various Feature Selection Techniques, in: Advanced Computing, Networking and Informatics-Volume 1, Springer, 2014, pp. 205– 211, doi:10.1007/978-3-319- 07353-8_24.
[12] C.Khammassi, S.Krichen, A galr wrapper approach for feature selection in network intrusion detection,Comput.Secur.70 (2017) 255–277, doi:10.1016/j.cose.2017.06.005.
[13] A. A. Diro and N. Chilamkurti, 'Distributed attack detection scheme using deep learning approach for Internet of Things', Future Gener. Comput. Syst., vol. 82, pp. 761-768, 2018.
[14] Sabeel U, Heydari SS, Mohanka H, Bendhaou Y, Elgazzar K, El- Khatib K (2019) Evaluation of deep learning in detecting unknown network attacks. In: 2019 international conference on Smart Applications, Communications and Networking, SmartNets 2019.
[15] Virupakshar KB, Asundi M, Channal K, Shettar P, Patil S, Narayan DG (2020) Distributed Denial of Service (DDoS) attacks detection system for OpenStack-based Private Cloud. Procedia Comput Sci 167:2297–2307.
[16] Asad M, Asim M, Javed T, Beg MO, Mujtaba H, Abbas S (2020) Deep-Detect: detection of Distributed Denial of Service attacks using deep learning. Comput J 63:983–994.
[17] Muraleedharan N, Janet B (2020) A deep learning based HTTP slow DoS classification approach using flow data. In: ICT Express.
[18] Sbai O, El Boukhari M (2020) Data flooding intrusion detection system for manets using deep learning approach. In: ACM international conference proceeding series. Association for Computing Machinery, New York, pp 281–286.
[19] Amaizu GC, Nwakanma CI, Bhardwaj S, Lee JM, Kim DS (2021) Composite and efficient DDoS attack detection framework for B5G networks. ComputNetw 188:107871.
[20] Cil AE, Yildiz K, Buldu A (2021) Detection of DDoS attacks with feed forward based deep neural network model. Expert Syst Appl 169:114520.
[21] Hai, T. H., &nam Huh, E. (2020). Network anomaly detection based on late fusion of several machine learning algorithms. International Journal of Computer Networks and Communications, 12(6), 117-131.
[22] Fernández A, Garcia S, Herrera F, Chawla NV. Smote for learning from imbalanced data: progress and challenges, marking the 15-year anniversary. J. Artif. Intell. Res. 2018; 61:863– 905.
[23] Tan X, Su S, Huang Z, Guo X, Zuo Z, Sun X, Li L. Wireless sensor networks intrusion detection based on smote and the random forest algorithm. Sensors 2019;19(1):203.
[24] Johnson JM, Khoshgoftaar TM. Survey on deep learning with class imbalance. J Big Data 2019;6:27. doi:10.1186/s40537-019- 0192-5.
[25] Karatas G, Demir O, Sahingoz OK. Increasing the performance of machine learning-based IDSs on an imbalanced and up-to-date dataset. IEEE Access 2020; 8:32150–62.

[26] Zhang H, Huang L, Wu CQ, Li Z. An effective convolutional neural network based on smote and gaussian mixture model for intrusion detection in imbalanced dataset. Comput. Netw. 2020; 177:107315.
[27] Maza, S., &Touahria, M. "Feature Selection Algorithms in Intrusion Detection System: A Survey," KSII Transactions on Internet and Information Systems, vol. 12, no. 10. Korean Society for Internet Information (KSII), 31-Oct-2018, doi:10.3837/tiis.2018.10.024.
[28] J. Mi, K. Wang, P. Li, S. Guo, Y. Sun, Software-defined green 5G system for big data, IEEE Commun. Mag. 56 (11) (2018) 116–123, doi:10.1109/MCOM.2017. 1700048.
[29] M. Abdullah, A. Balamash, A. Alshannaq, S. Almabdy, Enhanced intrusion detection system using feature selection method and ensemble learning algorithms, Int. J. Comput. Sci. Inf. Secur. (IJCSIS) 16 (2) (2018).
[30] Azhagusundari, and A.S. Thanamani, "Feature Selection based on Information Gain", International Journal of Innovative Technology and Exploring Engineering (IJITEE), Jan. 2013, pp 18- 21.
[31] A. Javaid, Q. Niyaz, W. Sun, and M. Alam, 'A deep learning approach for network intrusion detection system', in Proceedings of the 9th EAi International Conference on Bio-inspired information and Communications Technologies (formerly BJONETICS), 2016, pp. 21-26.
[32] Mera, C. and J.W. Branch, A survey on class imbalance learning on automatic visual inspection. IEEE Latin America Transactions, 2014. 12(4): p. 657-667.
[33] Wang, S., L.L. Minku, and X. Yao, A systematic study of online class imbalance learning with concept drift. IEEE transactions on neural networks and learning systems, 2018. 29(10): p. 4802-4821.
[34] Song, Q., Y. Guo, and M. Shepperd, A comprehensive investigation of the role of imbalanced learning for software defect prediction. IEEE Transactions on Software Engineering, 2018. 45(12): p. 1253-1269.
[35] Ivan, T. (1976). Two modifications of CNN. IEEE transactions on Systems, Man and Communications, SMC, 6, 769-772.
[36] G. Guo and N. Zhang, 'A survey on deep learning-based face recognition ', Comput. Vis. Image Underst., vol. 189, p. 102805, 2019.

BIOGRAPHIES OF AUTHORS

Hassan Afolabi received the Bachelor of Science (B.Sc.) degree in computer with electronics from the Lead City University, Ibadan Nigeria and Master of Science (M.Sc.) degree in Computer Science from the University of Ilorin, Nigeria. He is currently studying for his Ph.D. degree in computer engineering at the University of Kwazulu-Natal, Durban South-Africa. He is a member of the Nigeria Computer Society (NCS), Computer Professionals of Nigeria (CPN) and the Institute of Information Technology Professionals of South-Africa (IITPSA). His research areas of interest include Artificial Intelligence, Big Data Analysis, cybersecurity, deep learning and Internet of Things.

Prof. Dr. Abdurazzag Aburas received his bachelor's degree in Computer Sciences from Tripoli University, Tripoli-Libya in 1987. He obtained his master's degree in Computer & Information Technology and Ph.D. in Digital Image Processing from Dundee University and De Montfort University, the UK in 1993 and 1997, respectively. He worked in Jordan and UAE universities for five years, Electrical and Computer Engineering Department, Faculty of Engineering, International Islamic University Malaysia, and the International University of Sarajevo. He worked as a visiting full professor at Tecnológico de Monterrey, San Luis Potosi Campus, Mexico. At present, he is working at the University of KwaZulu Natal, School of Engineering, Howard College campus, Durban, South Africa. He has more than 75 publications in different international conferences and several research papers in international journals. He has research patents in the image processing field. His areas of research interest are Bigdata/Data Science, Machine Learning, Computer Security, Curriculum development, Digital Image Processing, and Human Mobile Interaction. He is a member of IEEE, HCI-UK, ARISE, IITPSA, and IMA Societies. He served as a member board of studies of Engineering Curriculum development (Reviewer and Improvement). He introduced new course curriculums such as Programming for Engineering (2008), Human Mobile Interaction (HMI) (2015), and Bigdata fundamentals (2017). He established Cloud Computing and Mobile Research Group (CCMRG) and Software Engineering Research Group (SERG) (2006-2009). At present, he is the coordinator of the Bigdata research group (UKZNBD). He has supervised several Ph.D. research degrees

and MSc research projects. He served as an external examiner for several Ph.D. and MSc. thesis assessors and currently supervising Ph.D. and MSc research projects at UKZN (2019-2022). He has published reference books based on Engineering Education (2013) and Human Mobile Interaction (2015) and his third book on Machine Learning for Engineering using Python in progress. He has more than 20 years of work worldwide in international Universities from 2000 to 2022. He is working in South Africa at UKZN at present time. He was a team leader to build a complete software package for the United Insurance Company in Tripoli-Libya from 1998-1999. The software package is still operating by the company until the present time. He has won several research awards in 2006, 2007, 2008, and 2019. His Research Interests are (but not limited to): Machine Learning, Data Science/Bigdata (Reduction not Compression), Cybersecurity, Digital Signal/Image Processing, Human Mobile Interaction (HMI), and Source Code Energy.