DDoSLSTM: Detection of Distributed Denial of Service Attacks on IoT Devices using LSTM Model

Vimal Gaur
Research Scholar, MMEC, Maharishi Markandeshwar Deemed to be University, Mullana (Ambala)-Haryana and
Reader, CSE Department
Maharaja Surajmal Institute of Technology (GGSIPU)
New Delhi, India
ORCID ID: 0000-0003-4097-1859
vimalgaur@msit.in

Rajneesh Kumar
Professor, Department of CSE, MMEC, Maharishi Markandeshwar (Deemed to be University), Mullana, Ambala - 133 207) ORCID ID: 0000-0002-8139-3533
drrajneeshgujral@mmumullana.org

2022 International Conference on Communication, Computing and Internet of Things (IC3IoT) | 978-1-6654-7995-0/22/$31.00 ©2022 IEEE | DOI: 10.1109/IC3IOT53935.2022.9767889

Abstract: **Distributed Denial of Service (DDoS) attack is a persistent complication in the network's security. These attacks have been detected by many machine learning algorithms and feature selection methods. This paper chose the Recurrent Neural Network based long short-term memory model that works on time series data and handles long time-dependent inputs, thereby detecting DDoS attacks. In our paper, we focused primarily on increasing the classification performance of the LSTM model. Multi-layer LSTM model has been used for binary and multiclass data and maximum accuracy attained is 99.46% (1- Layer LSTM with Binary data) followed by 99.16% for 2-Layer LSTM with Multiclass Grouped data. The proposed DDoSLSTM model outperforms other state-of-the-art techniques, including deep neural network (DNN), RNN, CNN, Transformers.**

Keywords: CICDDoS2019, DDoS, LSTM, Multiclass, RNN.

I. INTRODUCTION

Network traffic becomes more complicated and inconsistent, and DDoS attacks are expanding. DDoS attacks are one of the most devastating attacks. These attacks generate vague packets and disrupt the working of the target system by overwhelming it with a series of packets. Researchers proposed many machine learning classifiers for detecting DDoS attacks. Author [1] suggested the use of machine learning algorithms (RF, DT, XGBoost, SNN, DNN) for detecting attacks at an early stage. Further, they selected top features using different feature selection algorithms (Chi-Square, Extra Tree and ANOVA) and achieved 98.34% accuracy when XGBoost is coupled with ANOVA. According to our survey, deep learning algorithms characterize attacks automatically. DDoSTC is a hybrid neural network that combines transformers and a CNN to detect DDoS attacks on SDN [2]. Transformers are used for the classification of encrypted data. Authors in [3] calculated the area under curve value as 96.22% for the LSTM-FUZZY model on the CICDDoS2019 dataset. CyDDoS is another model proposed which implements DT, RF, GBoost, XGBoost, LightGBM, CatBoost and found maximum accuracy of 99.60% [4]. DDoSNet Model used RNN with autoencoder and found 99% accuracy for detecting reflection and exploitation attacks [5]. DeepSecure model enables detection and prediction of attacks [6]. It yields 99.70% as detection accuracy and 98.79% as prediction accuracy. DLSDN is a deep learning methodology for SDN networks and utilized stacked autoencoder Multi-layer Perceptron [7]. This gives an accuracy of 99.75%. Khemtech [8] proposed a hybrid method of DNN and LSTM for the partial dataset and achieved a 99.90% accuracy value. A new technique is proposed to improve the performance deterioration of deep learning techniques. This technique works explicitly on tiny samples of DDoS data [9]. A transferability metric has been designed to select the best network amongst the four networks in their work. The performance of the model drops initially on conducting a series of iterations performed using deep learning algorithms; later, it improves by 20.8%. A CNN based model is proposed to detect DDoS attacks [10]. In this work, CNN efficiently detects DDoS attacks by efficiently converting the network traffic dataset into image form. This methodology achieves an accuracy of 99.99 % with the binary classification of data.

In this paper, we presented a DDoSLSTM model for DDoS attack detection on the CICDDoS2019 evaluation dataset. The developed model consists of three classifications -: Binary, Multiclass ungrouped, Multiclass grouped.

The main contributions of this work are-:

1. We propose an RNN Based LSTM model to characterize attacks and hence detect DDoS attacks automatically.
2. DDoS attacks analysis is done on binary and multiclassification data.

The rest of the paper is organized as follows: Section 2 presents literature survey. Section 3 describes DDoSLSTM attack detection model for detecting DDoS attacks. Results and discussions are then described in Section 4. Section 5 presents conclusion.

II. LITERATURE SURVEY

Different researchers have proposed methodologies for the detection of DDoS attacks. These methodologies have been described in table 1.

Table I: PROPOSED METHODOLOGIES BY RESEARCHERS

Author	Model	Year	Methodology	Accuracy (%)
Wang et al. [2]	DDoSTC	2021	Transformers+ CNN	99.82 for 8:2 99.78 for 7:3 99.70 for 6:4 These results are for single-label data.
Novaes et al. [3]	LSTM-FUZZY	2020	LSTM	AUC=96.22
Lopes et al. [4]	CyDDoS	2021	(DT), random forest (RF), Gradient Boost (GBoost), Extreme Gradient Boosting (XGBoost), Light Gradient Boosted Machine (LightGBM), and CatBoost	99.6 (binary class).
Elsayed et al. [5]	DDoSNet	2020	RNN with Autoencoder	99.00
Kuadey et al. [6]	DeepSecure	2021	Detection Attack Model	99
			Slice Prediction Model	98
Ahuja et al. [7]	DLSDN	2021	SAE-MLP	99.75
Khempetch et al. [8]	-	2020	DNN+LSTM (Three DDoS attacks were discussed SYN, UDP, UDP-LAG)	99.90 (partial dataset).

III. DDoS ATTACK DETECTION MODEL USING DEEP LEARNING METHOD (RNN)

The steps to be followed in this model have been described below.

1. Dataset: CICDDoS2019 Evaluation dataset provided by Canadian Institute for Cybersecurity has been used for experimentation purposes.
2. Read Data: The Dataset has been processed on Jupyter Notebook and Google Colab with GPU acceleration.
3. Attribute Removal: Constant and Unique attributes have been removed to avoid overfitting.
4. Data Cleaning: Stratified sampling approaches have been deployed for proper data selection. Moreover, Benign label balancing has been performed.
5. Variable Standardization: Standard Scaler has been used to normalize the data.
6. Time-Series Based Sorting: The data has been sorted over timestamps to make it suitable for RNN based models.
7. Data Grouping: Grouping labels has been done to increase the model accuracy.
8. Data Splitting: 80:20 train-test ratio has been used for modeling the data.
9. Model Building: A multi-layer LSTM model with three dense layers has been built for data training.
10. Model Evaluation: The trained model has been validated against the validation set and tested over the training dataset.

Figure 1 depicts the above steps more clearly.

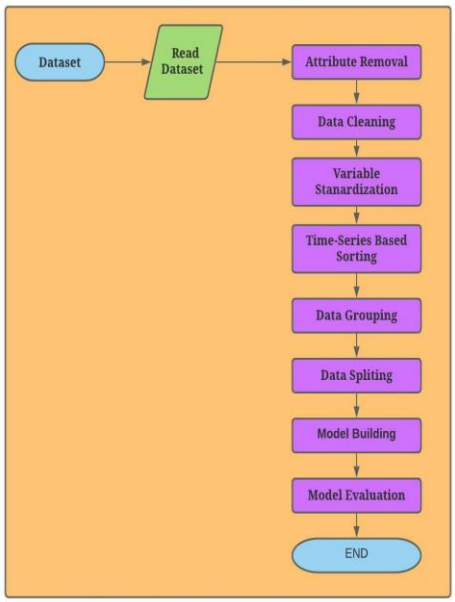


Fig. 1. DDoSLSTM Model

Model Building:

All the necessary libraries have been imported before model training. Thereafter, data has been loaded and a few columns Timestamp, SimilarHTTP, Flow Bytes/second, Flow Packets/second and WebDDoS have been dropped. This has been done to avoid overfitting, and afterward, it resulted in 77 features for modeling. Further, all the attacks have been labeled encoded based on their similarity with other attacks. Finally, StandardScaler has been employed to scale all the columns as it scales more than the maximum value in the training set. The final training dataset has been split into training and validation data by providing labels in the stratify parameter in the train_test_split function. RAM optimization is being performed throughout the process of deleting unwanted variables.

Since we have been working on time series data, the training and validation set has been sorted based on their index values. Then (X_train, Y_train) and (X_test, Y_test) have been prepared from training and validation sets, respectively. Next, a generator function (gen_sequence) has been prepared for changing the data and making it acceptable to the LSTM model. Finally, a windowing method has been employed for grouping data. The end of the window acts as the prediction parameter that needs to be predicted. Looping (type=For) is used with start and stop parameters. Start moves from the beginning of a set to end-sequence length (in this case is 500).

On the other hand, stop moves from (beginning + seq_length) to the end. Thirty elements have been skipped in between due to RAM limitations. gen_labels is used to transform target column values in a similar way as is done by gen_sequence. Later both the functions were called to form training and validation sets. Then they have been changed into NumPy arrays, and all the values have been converted into floating types. The final training and validation shapes for seq_array and val_seq_array are (22357, 500, 76) and (5577, 500, 76).

The dataset CICDDoS2019 has been classified into reflection-based and exploitation-based attacks [11]. The groupings of these attacks have been done on the primary machine learning methods.

IV. METHODOLOGY USED

All the experiments were operated on Google Colab with enabled GPU support (Intel UHD Graphics 620). This colab runs the program on an i5-8265U processor with 1.60 GHz clock speed and eight cores have been used with Samsung 512GB SSD. The experimental language was Python, with Tensorflow as a deep learning framework. The dataset has a time series-based component; therefore, an LSTM model has been employed [12]. A multi-layer LSTM model has been used [13-14]. The DDoS attack detection model has been trained with a learning rate of 0.0000001, adam optimizer, and 20 epochs. ReduceLROnPlateau has been employed to reduce overfitting with patience set to 2. Early Stopping has been used to find out the number of epochs that can be set for the model. Since the model architecture uses three dense layers, the following parameters have been used:

Learning rate= 0.0000001, Epochs= 20 with a callback function., Adam Optimizer, ReduceLROnPlateau Patience =2, number of units in the dense layer = 32, 16 and 8, the number of units in the LSTM layer = 50, 100, Batch Size=64, 128.

A deep learning neural network has been trained and tested on the partitioned data.

Scenario I

Binary Classification

Binary classifies whether an attack has occurred or not [15]. As clear from table 2, the 1-layer LSTM model achieves maximum accuracy of 99.46% in a minimum training time of 120 seconds. The best results have been achieved using the following configuration and depicted clearly using table 2:

Table II: PERFORMANCE OF LSTM MODEL USING BINARY CLASSIFICATION

Classification	LSTM Layers	LSTM Units	Dense Units	Total Training Time	Batch Size	Accuracy	Recall
Binary	1	50	32,16,8	120	128	0.9946	0.955
Binary	2	50	64,32,16	245	64	0.9945	0.955
Binary	3	50	64,32,16	1,134	64	0.9929	0.95

Input Layer
LSTM Units: 50
Batch Size: 128
Number of Epochs: 20
Learning Rate: 0.0000001
Fully connected Layer: 32 units
Fully connected Layer: 16 units
Fully connected Layer: 8 units
Output Layer: The neurons for the output layer is 1 for binary classification.

Scenario II
Multiclass Ungrouped Classification
In the second scenario, each category of attacks has been checked for accuracy. Maximum accuracy is achieved with a 2-layer LSTM model as 98.76% with 385 seconds as training time shown in table 3.

Table III: PERFORMANCE OF LSTM MODEL USING MULTICLASS UNGROUPED DATA

Classification	LSTM Layers	LSTM Units	Dense Units	Total Training Time	Batch Size	Accuracy	Recall
Multiclass Ungrouped	1	100	32,16,8	463	64	0.9862	0.944
Multiclass Ungrouped	2	50	32,16,8	385	64	0.9876	0.943
Multiclass Ungrouped	3	50	32,16,8	679	64	0.9811	0.943

The parameters used have been described below-:
Input Layer

LSTM Units: 50
Batch Size: 64
Number of Epochs: 20
Learning Rate: 0.0000001
Fully connected Layer: 32 units
Fully connected Layer: 16 units
Fully connected Layer: 8 units
Output Layer: The neurons for the output layer is 7 for ungrouped multi-classification problems.

Scenario III
Multiclass Grouped classification
In this paper, imbalanced class labels have been grouped into four labels as follows-:
Label 1: UDP, UDP-Lag, SYN (Reflection based attacks)
Label 2: NetBIOS, LDAP (Exploitation based attacks)
Label 3: BENIGN
Label 4: MSSQL
Hence maximum accuracy is achieved with 2-layer LSTM as 99.16% in 330 seconds shown in table 4.

Table IV: PERFORMANCE OF LSTM MODEL USING MULTICLASS GROUPED DATA

Classification	LSTM Layers	LSTM Units	Dense Units	Total Training Time	Batch Size	Accuracy	Recall
Multiclass Grouped	1	50	32,16,8	336	128	0.9907	0.9725
Multiclass Grouped	2	100	32,16,8	330	64	0.9916	0.975
Multiclass Grouped	3	50	64,32,16	1,597	64	0.9881	0.965

The parameters used have been described below-:
Input Layer
LSTM Units: 100 units
Batch Size: 64
Number of Epochs: 20
Learning Rate: 0.0000001
Fully connected Layer: 32 units
Fully connected Layer: 16 units
Fully connected Layer: 8 units

Output Layer: The neurons for the output layer is 4 for Grouped multi-classification problems. As a result, accuracy 99.46% (1- Layer LSTM with Binary data) > 99.16% (2- Layer LSTM with Multiclass Grouped data) > 98.76% (2- Layer LSTM with Multiclass Ungrouped data).

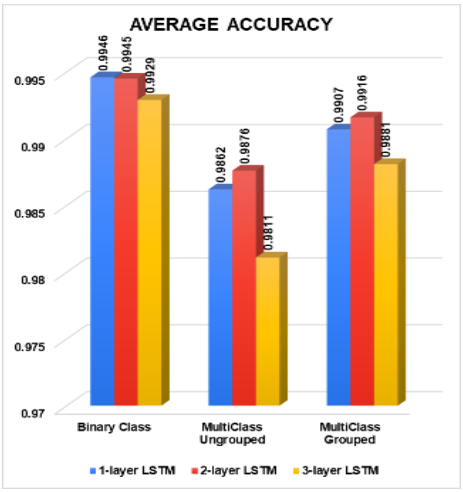
V. RESULTS AND DISCUSSIONS

The most common metrics used for model evaluation are Recall, Precision, Accuracy, F1-score. The performance parameters for the three scenarios have been evaluated in table 5.

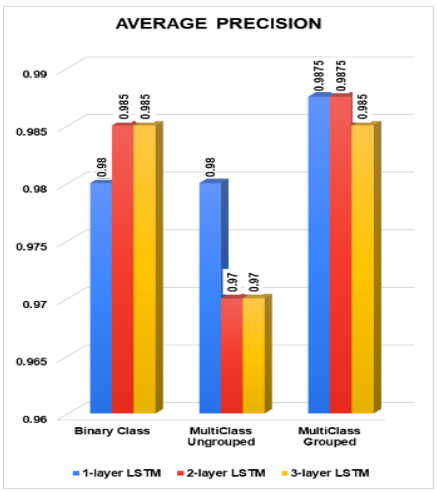
Table V: PERFORMANCE PARAMETERS OF DIFFERENT LAYERS WITH BINARY AND MULTICLASS DATA

	Recall			Precision			F1-Score			Accuracy		
Classification	**1-layer LSTM**	**2-layer LSTM**	**3-layer LSTM**	**1-layer LSTM**	**2-layer LSTM**	**3-layer LSTM**	**1-layer LSTM**	**2-layer LSTM**	**3-layer LSTM**	**1-layer LSTM**	**2-layer LSTM**	**3-layer LSTM**
Binary	0.955	0.950	0.955	0.980	0.985	0.985	0.970	0.970	0.955	0.9946	0.9945	0.9929
Multiclass Ungrouped	0.944	0.943	0.943	0.98	0.97	0.97	0.9585	0.9542	0.9557	0.9862	0.9876	0.9811
Multiclass Grouped	0.973	0.975	0.970	0.9875	0.9875	0.985	0.978	0.980	0.975	0.9907	0.9947	0.9811

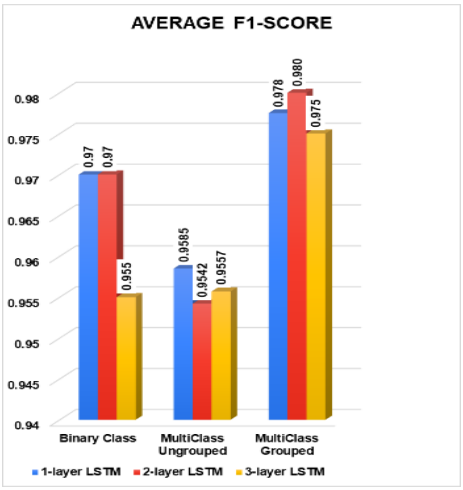
Maximum accuracy is obtained for the 2-layer LSTM model as 99.16% with multiclass grouped data. Multiclass grouping of data achieves a precision value of 98.75% with a 1-layer, 2-layer LSTM Model. The F1-Score is 98% for 2-layer LSTM multiclass grouped data. The maximum recall value for the 2-layer LSTM model with Multiclass grouped data is 97.5%. Experimental results show that LSTM with two-layer performs better than three-layer, accuracy decreases when the number of layers of the LSTM model increases from two to three. These results are clearly depicted in figure 2 (i), (ii), (iii) and (iv) respectively.



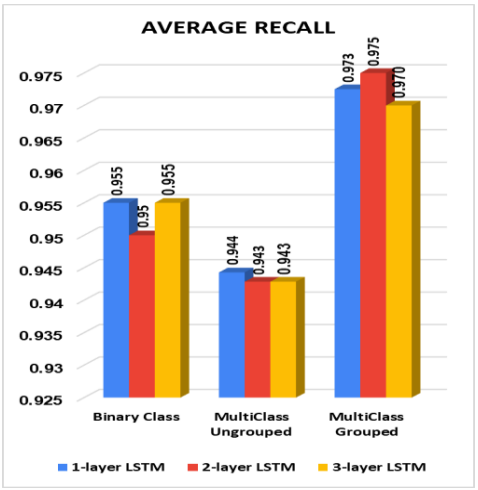
(i)



(ii)



(iii)



(iv)

Fig. 2: (i) Average Accuracy (ii) Average Precision (iii) Average F1-Score (iv) Average Recall for Multi-Layer LSTM

Table VI: COMPARISON WITH OTHER STATE-OF-THE ART MODELS

Author	Accuracy (%)
Gaur et al. [1]	98.34
Wang et al. [2]	99.82 for 8:2 99.78 for 7:3 99.70 for 6:4
Novaes et al. [3]	AUC=96.22
Lopes et al. [4]	99.6
Elsayed et al. [5]	99.00
Kuadey et al. [6] Model I	99.70
Kuadey et al. [6] Model II	98.79
Ahuja et al. [7]	99.75
Khempetch et al. [8]	99.90

Comparison with other state-of-the-art models has been made in Table 6. Thus it can be concluded that multiclass grouped classification attains maximum value for all performance parameters. Furthermore, the recently released dataset CICDDoS2019 has been adopted to identify various DDoS attacks.

VI. CONCLUSION

In this work, a Deep Learning model DDoSLSTM, is proposed to detect DDoS attacks. In the model, DDoS attack detection is treated on real-time data. One layer LSTM network shows good detection accuracy as 99.46% (binary data) followed by 99.16% for two-layered LSTM network (multiclass grouped data). Our model performs better than other state-of-the-art models with multiclass grouped data.

REFERENCES

1. V. Gaur, R. Kumar, “Analysis of Machine Learning Classifiers for Early Detection of DDoS Attacks on IoT Devices”, Arabian Journal of Science and Engineering, DOI: 10.1007/s13369-021-05947-3, 47(2), pp. 1353–1374, 2022.
2. H. Wang and W. Li,”DDosTC: A transformer-based network attack detection hybrid mechanism in SDN” in Sensors, vol. 21, issue 15, pp. 5047- 5057, 2021.
3. M. P. Novaes, L. F. Carvalho, J. Lloret and M. L. Proença, "Long Short-Term Memory and Fuzzy Logic for Anomaly Detection and Mitigation in Software-Defined Network Environment," in IEEE Access, vol. 8, pp. 83765-83781, 2020, doi: 10.1109/ACCESS.2020.2992044.
4. I. O. Lopes, D. Zou, F. A. Ruambo, S. Akbar, and B. Yuan. "Towards Effective Detection of Recent DDoS Attacks: A Deep Learning Approach." Security and Communication Networks 2021 (2021).
5. M. S. Elsayed, N.-A. Le-Khac, S. Dev, and A. D. Jurcut, “DDoSNet: a deep-learning model for detecting network attacks,” in Proceedings of the 2020 IEEE 21st International Symposium on “A World of Wireless, Mobile and Multimedia Networks” (WoWMoM), pp. 391–396, IEEE, Cork, Ireland, Aug. 2020.
6. N. A. E. Kuadey, G. T. Maale, T. Kwantwi, G. Sun and G. Liu, "DeepSecure: Detection of Distributed Denial of Service Attacks on 5G Network Slicing -Deep Learning Approach," in IEEE Wireless Communications Letters, doi: 10.1109/LWC.2021.3133479.
7. N. Ahuja, G. Singal and D. Mukhopadhyay, "DLSDN: Deep Learning for DDOS attack detection in Software Defined Networking," 2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence), 2021, pp. 683-688, doi: 10.1109/Confluence51648.2021.9376879.
8. T. Khempetch, P. Wuttidittachotti, “DDoS attack detection using deep learning,” in IAES International Journal of Artificial Intelligence; Yogyakarta, June 2021, Vol. 10, Iss. 2, pp. 382-388.
9. J. He, Y. Tan, W. Guo and M. Xian, "A Small Sample DDoS Attack Detection Method Based on Deep Transfer Learning," 2020 International Conference on Computer Communication

and Network Security (CCNS), 2020, pp. 47-50, doi: 10.1109/CCNS50731.2020.00019.

10. Hussain, S. G. Abbas, M. Husnain, U. U. Fayyaz, F. Shahzad and G. A. Shah, "IoT DoS and DDoS Attack Detection using ResNet," 2020 IEEE 23rd International Multitopic Conference (INMIC), 2020, pp. 1-6, doi: 10.1109/INMIC50486.2020.9318216.
11. I. Sharafaldin, A. H. Lashkari and S. Hakak, "Developing Realistic Distributed Denial of Service (DDoS) attack dataset and taxonomy," in 2019 International Carnahan Conference on Security Technology (*ICCST)*, Chennai, India, pp 1-8 Oct 2019.
12. S. Singh, S. V. Fernandes, V. Padmanabha and P. Rubini, "MCIDS-Multi Classifier Intrusion Detection system for IoT Cyber Attack using Deep Learning algorithm," 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV), 2021, pp. 354-360, doi: 10.1109/ICICV50876.2021.9388579.
13. N. Mishra and S. Pandya, "Internet of Things Applications, Security Challenges, Attacks, Intrusion Detection, and Future Visions: A Systematic Review," in IEEE Access, vol. 9, pp. 59353-59377, 2021, doi: 10.1109/ACCESS.2021.3073408.
14. K. Wehbi, L. Hong, T. Al-salah and A. A. Bhutta, "A Survey on Machine Learning-Based Detection on DDoS Attacks for IoT Systems," 2019 SoutheastCon, 2019, pp. 1-6, doi: 10.1109/SoutheastCon42311.2019.9020468.
15. V. Gaur and R. Kumar, "HCTDDA: Hybrid Classification Technique for Detection of DDoS Attacks," 2021 5th International Conference on Information Systems and Computer Networks (ISCON), 2021, pp. 1-5, doi: 10.1109/ISCON52037.2021.9702399.