IoT DoS and DDoS Attack Detection using ResNet

Faisal Hussain,Syed Ghazanfar Abbas
Al-Khawarizmi Institute of Computer Science (KICS) Lahore, Pakistan
faisal.hussain.engr@gmail.com, ghazanfar.abbas@kics.edu.pk

Muhammad Husnain,Ubaid U. Fayyaz
Al-Khawarizmi Institute of Computer Science (KICS) Lahore, Pakistan
muhammad.husnain@kics.edu.pk, ubaid@uet.edu.pk

Farrukh Shahzad,Ghalib A. Shah
Al-Khawarizmi Institute of Computer Science (KICS) Lahore, Pakistan
farrukh.shahzad@kics.edu.pk, ghalib@kics.edu.pk

***Abstract*—The network attacks are increasing both in frequency and intensity with the rapid growth of internet of things (IoT) devices. Recently, denial of service (DoS) and distributed denial of service (DDoS) attacks are reported as the most frequent attacks in IoT networks. The traditional security solutions like firewalls, intrusion detection systems, etc., are unable to detect the complex DoS and DDoS attacks since most of them filter the normal and attack traffic based upon the static predefined rules. However, these solutions can become reliable and effective when integrated with artificial intelligence (AI) based techniques. During the last few years, deep learning models especially convolutional neural networks achieved high significance due to their outstanding performance in the image processing field. The potential of these convolutional neural network (CNN) models can be used to efficiently detect the complex DoS and DDoS by converting the network traffic dataset into images. Therefore, in this work, we proposed a methodology to convert the network traffic data into image form and trained a state-of-the-art CNN model, i.e., ResNet over the converted data. The proposed methodology accomplished 99.99% accuracy for detecting the DoS and DDoS in case of binary classification. Furthermore, the proposed methodology achieved 87% average precision for recognizing eleven types of DoS and DDoS attack patterns which is 9% higher as compared to the state-of-the-art.**

***Index Terms*—Internet of Things, Convolution Neural Networks, ResNet, Intrusion Detection, IoT Attacks, DoS and DDoS Attack Detection.**

I. Introduction

Internet of Things (IoT) is the wireless interconnection of smart devices or things connected over the internet. In recent years, IoT has emerged as a promising technological solution for providing connectivity to myriads of heterogeneous devices across the globe. IoT can help us to access, control and manage these devices to get various functionalities in multiple application scenarios like smart home, smart healthcare, smart transportation, smart industry, etc. It can allow us to automate device control in order to facilitate the ease of device usage, to provide comfort and convenience to human being thus enhancing the overall quality of life.

In the current era, security is the major concern of IoT [1]. Denial of service (DoS) attacks and distributed denial of service (DDoS) attacks have been reported as the most common attacks on IoT devices and network [2]. A DoS attack is a malicious attempt done by an attacker using a single source to make a service or network resources inaccessible to legitimate users. When a DoS attack is launched using multiple distributed sources, it is called a DDoS attack. The DoS and DDoS attacks are increasing rapidly both in frequency and intensity with an average of 28.7K attacks per day [3], [4]. Recently, Neustar's report of cyber threats and trends [5] revealed that the DDoS attacks have been increased 200% in frequency while 73% increased in volume during the first six months of 2019 as compared to the same period in 2018 [6]. Fig. 1 depicts the surging trend of DDoS attacks as anticipated in Cisco's annual Internet report, 2018–2023 [7]. It can be observed that by 2023, the total count of DDoS attacks would become double, i.e., 15.4 million as compared to 2018. Hence, there is a crucial need for developing such solutions which can effectively detect and devastate the DoS and DDoS attempts.

So far, firewalls, intrusion detection systems (IDS) and intrusion prevention systems (IPS) are used as major security shields to protect the IoT devices and network from the cyber-attacks. However, the traditional firewalls, IDS and IPS cannot defend against the complex DDoS attacks [2], [8]–[10] as most of them filter the normal and suspicious traffic based upon the static predefined rules. However, the IDS and IPS that filter the intrusive attempts using artificial intelligence (AI) techniques are more reliable and effective as compared to the static predefined rules [8].

The traditional IDS use signatures or deep packet inspection (DPI) techniques for detecting malicious activities in the network. These techniques filter the packets based upon the packet content and header information. Unfortunately, such techniques have poor performance and become a bottleneck when deployed on high bandwidth and high-speed backbone links [8]. Moreover, these techniques fail to check packet contents when the encrypted traffic flows over the network [11]. Although many machine learning (ML) based solutions have been proposed for IoT attack detection, the prediction power of a well-tuned deep learning model especially convolutional neural network (CNN) is much better and effective as compared to the ML models [11], [12]. During the past few years, deep residual network (ResNet) drastically captivated the attention of researchers due to its tremendous performance [11]. Thereupon, in this work, we used ResNet [13] for IoT DoS and DDoS attack detection.

No matter, the deep learning models especially CNN models have achieved high significance due to their efficient performance in image processing and computer vision field [14]. However, these CNN models are also being used for detecting

2020 IEEE 23rd International Multitopic Conference (INMIC) | 978-1-7281-9893-4/20/$31.00 ©2020 IEEE | DOI: 10.1109/INMIC50486.2020.9318216

978-1-7281-9893-4/20/$31.00 ©2020 IEEE

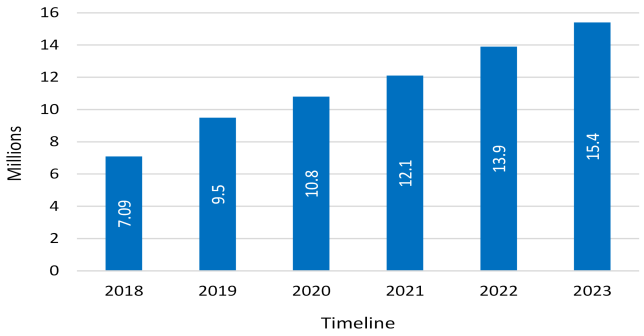

Fig. 1. Global Trend of DDoS Attacks 2018-2023 [7]

the network attacks. Liu *et al.* [11], proposed a CNN-based approach to detect the malicious traffic from NetFlow data. The authors first encoded the features then applied feature correlation and converted the data into images through surrounding correlation matrix. Finally, they fed these generated images to the deep learning models. Among these models, residual network (ResNet) [13] outperformed the other models. Likewise, Salman *et al.* [15] devised a framework for IoT device identification and attack detection. The authors used a self-generated dataset of seven IoT devices and evaluated the processed framework using two machine learning and three deep learning networks out of which a machine learning model, i.e., Random Forest outperformed. The authors in [12] revealed that ResNet [13] is prone to overfit in case of low dimensional and small size dataset due to which ResNet-based IDS do not perform well. To combat this challenge, the authors reconstructed the ResNet [13] model by simplifying the residual block. The experiments proved that the simplified ResNet performed better as compared to the actual ResNet [13] for low dimensional data.

The authors in [16] claimed that CNN best performs on images while the network traffic datasets are in non-image form. In order to efficiently use the potential CNNs for detecting the network intrusions, the authors proposed a methodology to convert the network traffic into a three-dimensional (3D) image. For this, the authors used a publicly available dataset, i.e., NSL-KDD dataset [17], applied fast Fourier transformation (FFT) onto it, converted it into 3D images and then passed it to a state-of-the-art CNN model to detect the network intrusions. Likewise, Li *et al.* [14] converted NSL-KDD dataset [17] feature values into binary vectors using a binary encoding scheme then transformed these vectors into images. These images were fed into two deep neural networks. The authors concluded that CNN models show better performance as compared to the machine learning methods.

Although the potential of CNN models is being used for developing intrusion detection systems, these CNN models do not perform efficiently when trained on non-image dataset [16]. Hence, there is a need for developing such a mechanism that transforms the network traffic into a representable form on which CNN models perform efficiently. Usually, the network traffic datasets are in low dimensional form, i.e., either in .pcap format or in .csv or .txt format. While the CNN models are designed and widely known for solving image processing and computer vision problems. Therefore, the CNNs especially, ResNet [13] do not perform well or overfit when trained on low dimensional and small-scale dataset [12]. In order to handle this issue, we proposed a methodology to convert the network traffic captures, i.e., non-image data into a representable form, i.e., image form and trained a state-of-the-art CNN model over the converted data in order to better detect the DoS and DDoS attack patterns.

The existing IoT attack detection systems are unable to detect the latest DoS and DDoS attacks [8]–[10]. The reason is that most of them are trained over either outdated datasets or the datasets used for training the proposed solutions do not include the modern reflective DDoS attacks like Network BIOS (NetBIOS) attack, Network Time Protocol (NTP) attack, Simple Service Discovery Protocol (SSDP) attack, UDP Lag (delay) attack, etc. In this work, we used the latest DDoS attack dataset, i.e., CICDDoS2019 [10] which contains 11 types of DoS and DDoS attacks and collected over a real-time network. Moreover, CICDDoS2019 [10] contains a large number of samples as compared to other network traffic datasets. In order to better detect the complex DoS and DDoS attacks, we used a state-of-the-art CNN model, i.e., ResNet [13] which showed efficient performance in detecting the image patterns. For getting the efficient performance of ResNet [13], we proposed a methodology to convert the non-image network traffic dataset into three-channel images.

A few works [11], [14], [16] converted the network traffic into images by applying some encoding method or some transformation technique like FFT to convert the data into image format. However, the proposed methodology is quite simple as we simply normalized the data instead of applying some computationally expensive encoding scheme or transformation technique like FFT. The proposed methodology showed better results for detecting the DoS and DDoS attack patterns as compared to the state-of-the-art work.

The rest of the paper is structured in the following manner: Section II defines the problem statement. Section III describes the proposed methodology for converting the non-image network traffic data into image form in order to better detect the DoS and DDoS attacks using a state-of-the-art CNN model. Section IV presents the results of the proposed methodology and shows how the proposed methodology outperforms state-of-the-art work. Lastly, Section V concludes the paper.

II. Problem Statement

DoS and DDoS attacks are the most common attacks in IoT [2]. The existing solutions are unable to detect the complex DoS and DDoS attacks [2], [8]–[10]. The reason is that most of them are trained over either outdated datasets or the datasets which do not include the modern reflective DDoS attacks like NetBIOS, NTP, SSDP, UDP Lag, etc., [10]. The potential of deep learning models especially convolutional neural networks

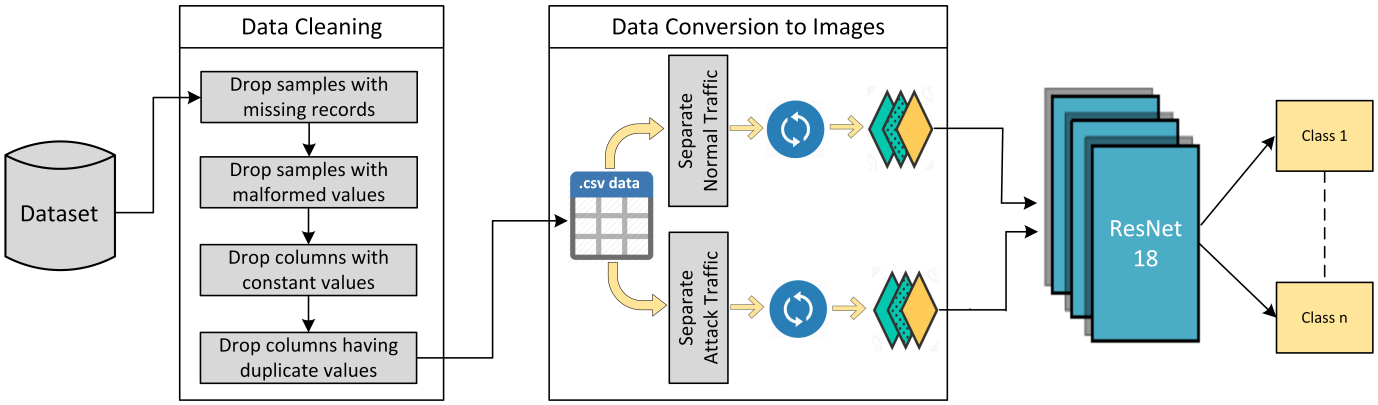

Fig. 2. Proposed Methodology For Detecting IoT DoS and DDoS Attacks using ResNet18

(CNNs) is being used for detecting the network attacks. However, CNNs do not perform efficiently when trained on non-image or low dimensional dataset [12], [16] since CNNs are specially designed to solve computer vision problems. The network traffic datasets exist either in .pcap format or in .csv or .txt format. Therefore, to efficiently utilize the potential of CNNs for the DoS and DDoS attacks detection, the network traffic data should be converted into image form.

III. Proposed Methodology

The proposed methodology consists of four key steps which include: data acquisition, data cleaning, data conversion and attack pattern recognition. Fig. 2 provides an overview of the proposed methodology. The first step of the proposed methodology is to acquire network traffic data. Once the data is acquired, it is preprocessed in order to get the refined data. During the preprocessing, we will perform two major steps, i.e., data cleaning and conversion of cleaned data into three-channel images. The final step is to train and test the CNN model over the preprocessed data in order to evaluate the performance for detecting the DoS and DDoS attack patterns. All these steps are explained in the following subsections.

A. Data Acquisition

The data acquisition is the first step of the proposed methodology to acquire both normal and attack traffic. Generating extensive normal and attack traffic by setting up a real-time network is an onerous task as it requires significant network resources, diversity of network normal and attacks traffic captures, etc. Moreover, it is also a time and money consuming process to set up a huge network. However, one can get rid of this laborious job by using a publicly available network traffic dataset. In order to get a quality dataset, we analyzed some publicly available datasets based upon the following criteria:

- The dataset must consist of real-time network traffic.
- The dataset must be extensive and versatile.
- The dataset must comprise of the latest DoS and DDoS attacks.
- The dataset should cover a variety of attack vectors.

We reviewed some publicly available datasets which include KDD-99 [18], NSL-KDD [17], DEFCON [19], CAIDA [20], UNSW-NB15 [21] and CICDDoS2019 [10]. Based on the above-mentioned criteria, for this work, we selected CICDDoS2019 [10] dataset. CICDDoS2019 [10] is the latest dataset which contains a large number of samples as compared to the other network traffic datasets. Moreover, it contains both inbound and outbound traffic of the latest DoS and DDoS attacks while most of the above-mentioned datasets are either outdated or contain limited attack scenarios of DoS attacks [10]. Furthermore, the CICDDoS2019 [10] dataset contains more than 80 network flow-related features and eleven types of latest DoS and DDoS attacks traffic collected over a real-time network. The details of these features are described in [22].

B. Data Preprocessing

Once the data is acquired, the next stage is to preprocess the data in order to bring it in a refined form. During this stage, we performed three major steps, i.e., data cleaning, data conversion, and train test and validation split.

1) Data Cleaning: The acquired network traffic data was in .csv format which includes more than 80 flow features. In order to better train our model for attack pattern detection, we removed the unwanted features from the data set which are not useful for classifying the attack and normal traffic. These features include Flow ID, Source IP, Source Port, Destination IP, Destination Port, Protocol and Timestamp. As based upon these static features, one cannot decide whether a certain flowid, srcIP, etc., whenever found will always generate malicious or normal packets. That's why we dropped such unwanted features and excluded them from our training set.

Thereafter, we analyzed the whole dataset in order to deal with missing or malformed data. For this purpose, we first checked that which samples contain missing values, which samples have inadequate values like nan, -inf, +inf, etc. As we had a large number of samples in the dataset, so we dropped all those samples which comprise of missing or malformed values.

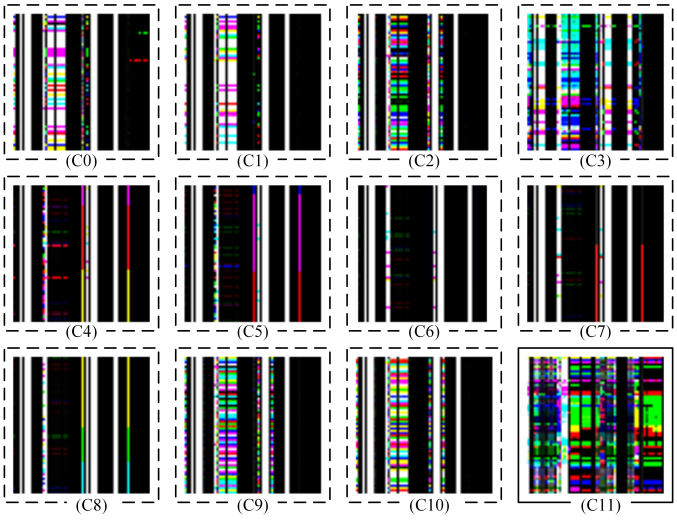

Fig. 3. Images obtained during Data Conversion Stage. (C0): Syn attack, (C1): TFTP attack, (C2): UDP Lag attack, (C3): DNS attack, (C4): LDAP attack, (C5): MSSQL attack, (C6): NetBIOS attack, (C7): NTP attack, (C8): SNMP attack, (C9): SSDP attack, (C10): UDP attack, (C11): Normal traffic

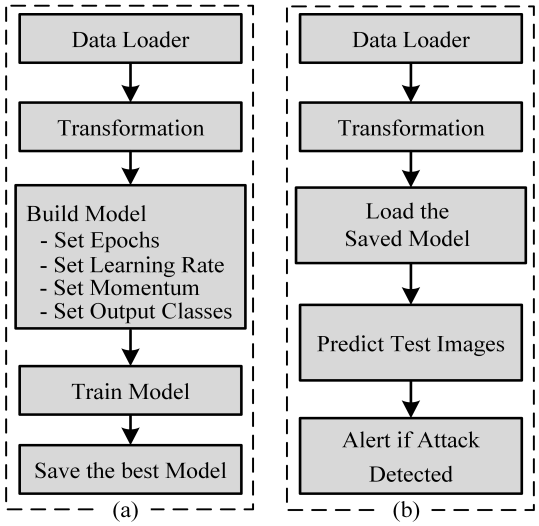

Fig. 4. Steps for (a) Training and (b) Testing ResNet18 Model

After that, we figured out the features which were either duplicate or entirely had a constant value in case of all labels. Such constant features are not useful for discriminating the attack or normal traffic and may decrease the performance of the machine learning model, if included in the training set. Therefore, we also dropped constant features from the training set. These features include Bwd PSH Flags, Fwd URG Flags, Bwd URG Flags, FIN Flag Count, PSH Flag Count, ECE Flag Count, Fwd Avg Bytes/Bulk, Fwd Avg Packets/Bulk, Fwd Avg Bulk Rate, Bwd Avg Bytes/Bulk, Bwd Avg Packets/Bulk and Bwd Avg Bulk Rate. On the other hand, the duplicate features are those which have similar values but has a different name. In the case of duplicate features, we keep the first original feature and dropped its duplicate feature. These features include RST Flag Count, Fwd Header Length, Subflow Fwd Packets, Subflow Fwd Bytes, Subflow Bwd Packets and Subflow Bwd Bytes. Finally, after the cleaning the data we left with 60 features which were unique and important.

2) Data Conversion: As mentioned earlier that CNN performs well when trained on an image-based dataset. Since the network traffic dataset is captured in non-image format, i.e., it can be either in .csv file, .txt file, or .pcap file. So, in order to get efficient results for network attack detection, we need to convert the network traffic data into image form. For this purpose, we normalized the dataset w.r.t each feature using (1).

$$X' = \frac{X - Min(X)}{Max(X) - Min(X)} \times 255 \quad (1)$$

In order to convert the network traffic into image form, we first separated the all normal and attack samples into two data frames as shown in Fig. 2. After that, we selected the chunk of 180 samples iteratively, converted them into an image of shape 60x60x3 in such a way that first 60 samples of each chunk were converted into image matrix of channel 1, next 60 samples of each chunk were converted into image matrix of channel 2 and the last 60 samples of each chunk were converted into image matrix of channel 3 and finally, mapped these matrices into RGB channels of an image. We converted these normal and attack samples matrices to images using the OpenCV library and labelled them accordingly. The same procedure was followed for each .csv file of the dataset until all the samples were converted into images. Fig. 3 presents one sample of the converted images from each class, i.e., C0 to C11 which includes the 11 types of DoS and DDoS attacks along with normal traffic. In Fig. 3, the images enclosed inside a dot line box exhibits the attack images while the image inside the solid line box represents the normal image.

3) Train and Test Split: After converting the dataset into image form, the next step is to divide the dataset into a training set, validation set and testing set. In this regard, we randomly selected 2500 images from each category for testing and the images left were used for training the CNN model. Some of the classes had less than 2500 images, so we included all of them in the testing set.

C. Attack Detection

Once the data is organized into train, test and validation set, the next step is to pass this data to the CNN model so that the model train itself on the given data, learn the attack and normal traffic patterns and validate itself. We used ResNet18 [13] model which consists of 18 layers out of which 10 are convolution layers and 8 pooling layers.

Fig. 4 – (a) shows the steps done for training the ResNet18 [13] model on the network traffic dataset. We first loaded the train, validation set. The ResNet [13] is designed to accept the images with size 224 x 224 [11] while the preprocessed images had dimension 60 x 60 x 3. So, we transformed the preprocessed images into 224 x 224 x 3. After transforming the images, the next step is to build the model.

		Predicted Class											
Actual Class		C0	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11
	C0	1799	0	413	0	0	0	254	0	22	11	0	1
	C1	0	1316	0	0	0	137	1	0	0	7	1	0
	C2	2	2	0	0	0	0	1030	0	1	2	0	0
	C3	0	3	0	2474	9	2	1	2	7	2	0	0
	C4	0	0	0	9	2234	127	0	0	125	5	0	0
	C5	0	1	0	0	2	2494	3	0	0	0	0	0
	C6	0	3	0	0	0	9	2486	0	2	0	0	0
	C7	0	5	1	0	0	0	0	2491	3	0	0	0
	C8	0	3	0	0	0	4	116	0	2370	7	0	0
	C9	0	61	12	0	0	2	0	0	19	2291	115	0
	C10	0	114	37	0	0	0	0	0	3	518	1828	0
	C11	27	34	90	0	0	1	17	148	7	12	6	2158

Fig. 5. Confusion Matrix obtained using Trained ResNet Model for DDoS Attack Detection

For building the ResNet [13] model, we have to set some parameters as per use case. The original ResNet [13] model had 1000 output classes but, in our use case, we set output class as 1 for binary classification while for multi-class classification, we set output classes as 12. So, we changed the last layer of ResNet [13] model, to predict the given image according to our use case. Similarly, we also need to set some other parameters while building a model which includes epochs, learning rate, momentum and optimizer. We set the Resnet18 [13] model with leaning rate 0.0001 with the momentum of 0.9, iterate over 10 epochs for binary classification and 50 epochs for multi-class classification with Stochastic gradient descent (SGD) optimizer. After building the models, we started the training of ResNet [13] model over the prescribed parameters. While training the model, after each epoch we evaluated the model performance and saved the model weights if it gives the best accuracy. This evaluation process continued until the last epoch. Finally, we come up with the two best models, one for binary classification and other for multi-class classification.

Once the trained model with the best accuracy is saved, thenceforth, we need to test the trained models over the test set which remained unrevealed during the training phase. Fig. 4 – (b) illustrates the steps performed for testing the trained model. For testing the saved model, we first loaded the test images then transformed them into the dimension of 224 x 224 x 3 in a similar manner that we followed in the training phase. Afterwards, we loaded the saved models and passed the transformed images to it, so that it predicts whether the given images are normal or malicious in case of binary classification and if they are malicious then also predict their attack class in case of multi-class classification. Lastly, the predicted labels were compared with the actual labels in order to measure the performance of the trained model. The following section addresses the results achieved during the training and testing phase.

TABLE I
RESULTS COMPARISON

Method	Precision	Recall	F1-Measure
Sharafaldin *et al.* [10]	0.78	0.65	0.69
Proposed	0.87	0.86	0.86

IV. RESULTS AND DISCUSSION

The performance of the proposed methodology is evaluated based on four commonly used performance metrics which include precision, recall, accuracy and F1-measure. In case of multi-class classification, all these parameters are calculated individually for each class from the confusion matrix which is shown in Fig. 5, then the average of each parameter is included in Table I. These parameters are defined as:

Precision - It defines the ratio truly detected attacks and all packets that are classified as attacks. Mathematically, it is expressed in (2):

$$Precision = \frac{TP}{TP + FP} \times 100 \tag{2}$$

Recall - It is the ability of the system to correctly detecting the attack upon the occurrence of the security breach. It is also called as the true positive rate. Mathematically, it is described in (3)

$$Recall = \frac{TP}{TP + FN} \times 100 \tag{3}$$

Accuracy - It is defined as the ability of the system to correctly classify the attack packet as an "attack packet" and normal packet as a "normal packet". It tells about the ratio of correct predictions with respect to all samples. Mathematically, it is expressed in (4):

$$Accuracy = \frac{TP + TN}{TP + FN + TN + FP} \times 100 \tag{4}$$

F1-Score - It is defined as the harmonic mean of precision and recall. Mathematically, it is represented in (5):

$$F1\text{-}Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{5}$$

We evaluated the proposed methodology for DoS and DDoS attack detection based on the above-mentioned parameters during the training and testing phase for both detecting and recognizing the inbound and outbound DoS and DDoS attacks in IoT networks. In case of binary classification, the proposed methodology achieved 99.99% accuracy for detecting the DoS and DDoS attacks. While in the case of multi-class classification, the proposed methodology achieved 87.06% accuracy. Fig. 6 illustrates the percentage of correctly predicted attacks of 11 classes. It can be observed that Syn (C0), DNS (C3), LDAP (C4) attacks and normal traffic (C11) are detected with the highest precision, while UDP Lag attack is misdetected as NetBIOS attack. However, if we look at it in binary class perspective, then UDP Lag attack is correctly predicted as an attack.

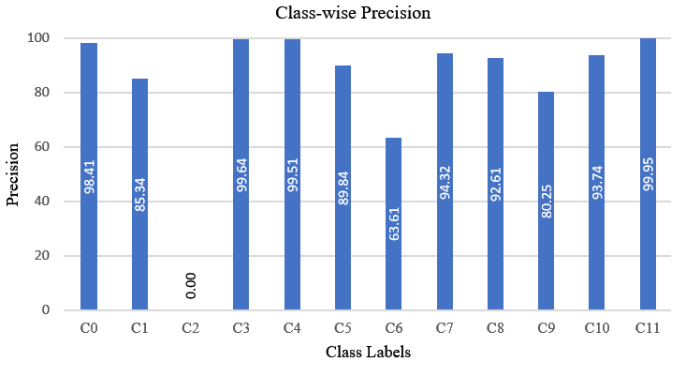

Fig. 6. Class-wise Precision obtained using Trained ResNet18 Model for DDoS Attack Detection

Overall, the proposed methodology using ResNet-18 [13] model for multi-class classification, showed 88.56% accuracy with a loss of 0.386 during the training phase. While upon testing, it showed the accuracy of 87.06%. We also compared the results of our proposed methodology with a state-of-the-art solution in Table I. It can be noticed that for detecting DoS and DDoS attack patterns, the proposed methodology exhibited 9% more precision as compared to the state-of-the-art solution. Furthermore, it achieved 21% higher recall, i.e., true positive rate and 17% higher F1-score as compared to the state-of-the-art solution which was proposed on the same dataset.

V. Conclusion

The recent cyber-attack statistics reveal that denial of service (DoS) and distributed denial of service (DDoS) are the most occurring attacks in IoT networks and devices which are rising both in frequency and intensity by lapse of time. The convolutional neural network (CNN) models have gained a lot of significance in image classification tasks due to their outstanding performance. However, these models do not perform well when trained on a non-image dataset as they are designed to find the patterns from images. In order to use the potential of CNN models, in this work, we proposed a methodology to convert the non-image network traffic dataset into three-channel image form. Thereafter, we trained a state-of-the-art CNN model, i.e., ResNet over the transformed dataset and analyzed its performance for detecting the latest DoS and DDoS attacks. The proposed methodology is simple as compared to the existing works in such a way that it only normalizes the features and does not use any encoding scheme or transformation technique like fast Fourier transformation (FFT) to convert the preprocessed data into images. The proposed methodology achieved 99.99% accuracy for detecting DoS and DDoS attacks. Furthermore, it attained 87% precision for recognizing the 11 types of DoS and DDoS attacks which is 9% higher as compared to the state-of-the-art.

References

[1] E. Hossain, I. Khan, F. Un-Noor, S. S. Sikander, and M. S. H. Sunny, "Application of big data and machine learning in smart grid, and associated security concerns: A review," *IEEE Access*, vol. 7, pp. 13 960–13 988, 2019.

[2] T. G. Nguyen, T. V. Phan, B. T. Nguyen, C. So-In, Z. A. Baig, and S. Sanguanpong, "Search: A collaborative and intelligent nids architecture for sdn-based cloud iot networks," *IEEE access*, vol. 7, pp. 107 678–107 694, 2019.

[3] *What Is a DoS Attack?*, (accessed January 11, 2020). [Online]. Available: https://www.datto.com/library/what-is-a-dos-attack

[4] M. Jonker, A. King, J. Krupp, C. Rossow, A. Sperotto, and A. Dainotti, "Millions of targets under attack: a macroscopic characterization of the dos ecosystem," in *Proceedings of the 2017 Internet Measurement Conference*. ACM, 2017, pp. 100–113.

[5] *Neustar Cyber Threats and Trends Report Q1 2019*, (accessed January 11, 2020). [Online]. Available: https://www.discover.neustar/rs/717-IIA-274/images/Neustar%20Cyber%20Threats%20and%20Trends%20Report%20Q1%202019%20-%20Web.pdf

[6] *DDoS attack statistics and facts for 2018-2019*, (accessed January 11, 2020). [Online]. Available: https://www.comparitech.com/blog/information-security/ddos-statistics-facts/

[7] *Cisco Annual Internet Report (2018–2023) White Paper*, (accessed June 11, 2020). [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html

[8] B. A. Khalaf, S. A. Mostafa, A. Mustapha, M. A. Mohammed, and W. M. Abduallah, "Comprehensive review of artificial intelligence and statistical approaches in distributed denial of service attack and defense methods," *IEEE Access*, vol. 7, pp. 51 691–51 713, 2019.

[9] S. Ghazanfar, F. Hussain, A. U. Rehman, U. U. Fayyaz, F. Shahzad, and G. A. Shah, "Iot-flock: An open-source framework for iot traffic generation," in *2020 International Conference on Emerging Trends in Smart Technologies (ICETST)*. IEEE, 2020, pp. 1–6.

[10] I. Sharafaldin, A. H. Lashkari, S. Hakak, and A. A. Ghorbani, "Developing realistic distributed denial of service (ddos) attack dataset and taxonomy," in *2019 International Carnahan Conference on Security Technology (ICCST)*. IEEE, 2019, pp. 1–8.

[11] X. Liu, Z. Tang, and B. Yang, "Predicting network attacks with cnn by constructing images from netflow data," in *2019 IEEE 5th 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)*. IEEE, 2019, pp. 61–66.

[12] Y. Xiao and X. Xiao, "An intrusion detection system based on a simplified residual network," *Information*, vol. 10, no. 11, p. 356, 2019.

[13] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[14] Z. Li, Z. Qin, K. Huang, X. Yang, and S. Ye, "Intrusion detection using convolutional neural networks for representation learning," in *International Conference on Neural Information Processing*. Springer, 2017, pp. 858–866.

[15] O. Salman, I. H. Elhajj, A. Chehab, and A. Kayssi, "A machine learning based framework for iot device identification and abnormal traffic detection," *Transactions on Emerging Telecommunications Technologies*, p. e3743, 2019.

[16] W. Liu, X. Liu, X. Di, and H. Qi, "A novel network intrusion detection algorithm based on fast fourier transformation," in *2019 1st International Conference on Industrial Artificial Intelligence (IAI)*. IEEE, 2019, pp. 1–6.

[17] *NSL-KDD dataset*, (accessed January 11, 2020). [Online]. Available: https://www.unb.ca/cic/datasets/nsl.html

[18] *KDD Cup 1999 Data*, (accessed January 11, 2020). [Online]. Available: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

[19] *DEFCON*, (accessed January 11, 2020). [Online]. Available: https://www.defcon.org/html/links/dc-ctf.html

[20] *Center for Applied Internet Data Analysis (CAIDA)*, (accessed January 11, 2020). [Online]. Available: https://www.caida.org/data/

[21] N. Moustafa and J. Slay, "Unsw-nb15: a comprehensive data set for network intrusion detection systems (unsw-nb15 network data set)," in *Military Communications and Information Systems Conference (MilCIS), 2015*. IEEE, 2015, pp. 1–6.

[22] *NETWORK TRAFFIC FLOW ANALYZER*, (accessed January 11, 2020). [Online]. Available: http://www.netflowmeter.ca/netflowmeter.html