Intelligent Automation & Soft Computing
DOI:10.32604/iasc.2022.024668
Article

Tech Science Press

Ensemble Deep Learning Models for Mitigating DDoS Attack in Software-Defined Network

Fatmah Alanazi*, Kamal Jambi, Fathy Eassa, Maher Khemakhem, Abdullah Basuhail and Khalid Alsubhi

Department of Computer Science, King Abdulaziz University, Jeddah, Saudi Arabia
*Corresponding Author: Fatmah Alanazi. Email: fatmah.rtian@gmail.com

Received: 27 October 2021; Accepted: 30 November 2021

Abstract: Software-defined network (SDN) is an enabling technology that meets the demand of dynamic, adaptable, and manageable networking architecture for the future. In contrast to the traditional networks that are based on a distributed control plane, the control plane of SDN is based on a centralized architecture. As a result, SDNs are susceptible to critical cyber attacks that exploit the single point of failure. A distributed denial of service (DDoS) attack is one of the most crucial and risky attacks, targeting the SDN controller and disrupting its services. Several researchers have proposed signature-based DDoS mitigation and detection techniques that rely on manually configuring the policies. As the massive traffic from heterogeneous networks increases, conventional solutions are ineffective due to the lack of automation and human interference. This necessitates producing a detection solution, more effective than traditional ones, to ensure SDN security, resiliency, and availability. This paper addresses this problem by proposing a deep learning (DL)-based ensemble solution for efficient DDoS attack detection in SDN. Four hybrid models are presented by adopting three ensemble techniques and different DL architectures, namely convolutional neural network, long short-term memory, and gated recurrent unit, to improve the SDN traffic classification. The experimentation was conducted on the benchmark flow-based dataset CICIDS2017. High detection accuracy (99.77%) with a small number of flow-based features was achieved by our ensemble model, as our experimental results will demonstrate. The proposed solution was evaluated by several standard assessment matrices and by comparing against other state-of-the-art algorithms from the network security literature.

Keywords: Distributed denial of service; anomaly detection; software-defined network; deep learning; convolutional neural network

1 Introduction

The advancement in network technology has changed how networking works for the Cloud, the Internet of Things, and the Internet through simplifying its management and control. A software-defined network (SDN) is a new form of network that permits the detachment of control and data planes in addition to the centralization of control in one place. This separation offers more flexibility for the network administration and effective usage of network resources [1]. The control plane helps secure the network by producing a global overview of the network state. However, it drives several challenging issues such as scalability, reliability, and vulnerability to many potential network security attacks [2].

Distributed denial of service (DDoS) attacks on SDN can disturb the network's service availability by targeting different layers in SDN, specifically, the application, control, and physical layers [3]. DDoS attacks deny legitimate clients from reaching the services or resources provided by the network. This is typically accomplished by exhausting the target with extra requests that overload the system and affect legitimate request processing. It is known that the controller's resources such as memory, CPU, and bandwidth are limited. Hence, processing a large number of actions set for each "Packet In" request coming from the data plane overwhelms the resources of the controller. Fig. 1 describes the DDoS scenario targeting the controller. As reported by AWS Shield, a DDoS attack reached 2.3 Tbps in the first quarter of 2020 [4]. The distributed nature of DDoS attacks makes launching the attack an easy process, whereas the defending mechanisms are challenging. As sophisticated variants of DDoS attacks are developed, more than conventional detection approaches are needed to tackle these attacks [3]. The main challenges in the detection of DDoS in SDN traffic are the lack of labeled network datasets, dynamic and diverse nature of traffic [5], the similarity of normal and malicious packets in DDoS attacks [6], and switch layer hiding of packets [7]. Detecting the DDoS can be done by several methods, such as statistical, data mining, and machine learning methods. SDN provides an opportunity for leveraging artificial learning approaches in support of automated attack mitigation, particularly beneficial for complex, large-scale, and highly dynamic environments [8]. Recently, deep learning (DL) approaches are becoming more popular and efficient to build attack detection systems with high accuracy and low false alarm rate. In this paper, the following contributions are presented:

- We introduce a DL-based DDoS attack detection solution in the SDN by applying ensemble techniques.
- In our work, new combinations of most classical DL models, such as convolutional neural network (CNN), gated recurrent unit (GRU) and long short-term memory (LSTM), are used to propose a model with the best accuracy and false positive rate (FBR).
- We evaluate the proposed models using the flow-based published state-of-the-art dataset CICIDS2017. Through experiments, a detection rate of 99.77% was achieved by one of our proposed ensemble models using limited raw features.
- Our solution can be applied in detecting a couple of further attacks by restructuring the CNN and RNN models based on the selected features.

The remainder of this paper is organized as follows: In Section 2, we briefly introduce a literature review. Section 3 contains the methodology and some background details about DL algorithms used; additionally, it describes the CICIDS20127 dataset and its features. In Section 4, we give a description of our proposed approach for DDoS attack detection. Section 5 shows the experimental results and performance evaluation of our approach. Finally, the conclusion of our study and future prospects are presented in Section 6.

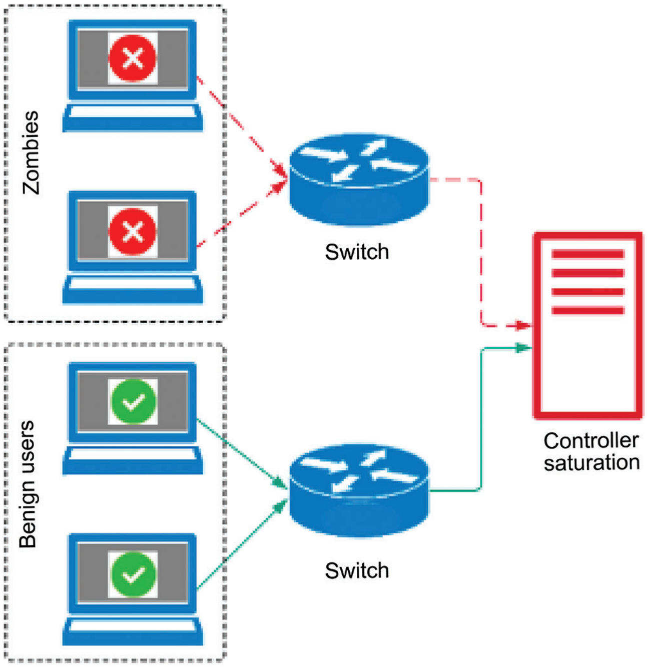


Figure 1: DDoS targeting the SDN controller

2 Related Works

With the advent of emerging wireless networks such as the Internet of Things (IoT) and 5G, the future networks, SDNs, are more exposed to vulnerable threats and cyber attacks. DDoS is considered one of the most widely and largely attacked. Based on a comprehensive consideration of the next generation of intrusion detection and SDNs concerned with research, applying machine learning techniques has strong potential solutions in anomaly detection. Most importantly, DL strategies are outperforming traditional learning strategies and have the option to convey higher detection accuracy. Moreover, the SDN paradigm with having a centralized control plane that can monitor the network states globally has shown very promising results in DDoS detection [9].

A naïve Bayes (NB) classification algorithm is proposed by the researchers for detecting DDoS attacks [10]. The framework developed by the paper is based on a multi-SDN controller environment and is validated on an NSL-KDD dataset. The results show that the proposed technique achieves an accuracy of 98%.

Tang et al. [11] proffer an intrusion detection system (IDS) based on DL under the context of SDN to detect anomalies using (GRU). The authors used NSL-KDD and CICIDS2017 datasets for model evaluation and achieved an accuracy of a maximum of 89% and 99%, respectively. However, NSL-KDD is a packet-based dataset and not suitable for SDN. Similarly, Dey et al. [12] proposed a GRU-LSTM-based model with varied feature selection methods for flow-based malicious detection in SDN networks. Recursive feature elimination and the ANOVA F-test were applied to select the feature from the NSL-KDD dataset. However, the detection accuracy obtained by the model was not high enough.

As of late, ensemble solution has been utilized majorly for research for anomaly detection in SDNs. The ensemble performs better as compared to individual models, thus decreasing overfitting issues, bias, and variance while improving predictions. A DDoS detection architecture was proposed by Haider et al. [13] through different ensemble DL-based models, including CNN, LSTM, and RNN. The researchers achieved 99.45% attack detection accuracy by performing ensemble CNN model construction using two similar CNN models and Keras merging layers. In the same context, an ensemble CNN-based model was applied in another work by Haider et al. [14] for DDoS attack detection using the CICIDS2017 dataset for validation and achieved a maximum accuracy of 99.48%. However, the experimental results present significantly high computational complexity.

Reference [15] proposes an ensemble model based on several ML classifiers, where each classifier can detect specific types of attacks and provides a more intelligent and robust framework for threat detection. Moreover, the framework also incorporated a technique known as a reduced feature set for improving the accuracy. The experimental analysis based on the NSL-KDD dataset shows an accuracy of 99.1% and can even perform with new emerging DDoS attacks. A voting-based ensemble technique is proposed for DDoS detection in [16]. The authors analyzed three different ensemble methods—Voting-RKM, Voting-CKM, and Voting-CMN—with standard datasets to propose a high-performance ensemble model. The simulation results show that the Voting-RKM ensemble technique achieves the highest accuracy compared to other methods.

In a multiclass classification, Sharma et al. [17] presented a real-time IDS, which was based on a multilayer and ensemble model to detect specific attack types. A binary classifier layer was used to decrease the complexity of multiclass issues. Each extreme learning machine (ELM) represents a binary classification for each intruder and runs independently, while their different outcomes are fed to the last layer. In the final stage, the author proposed a SoftMax stage to generate probabilistic results for each attack class.

Another hybrid approach was introduced by Mhamdi et al. [18], which combined an autoencoder with one-class SVM (OC-SVM) for anomaly detection. The stack autoencoder (SAE) fed OC-SVM with low dimensional representation inputs to classify the traffic into legitimate or anomaly data. This approach shows promising detection accuracy, but their false positive rate is still high. In a different study, a hybrid approach for multiclass classification was used in an intrusion detection environment. Reference [19] used an improved conditional variational autoencoder (ICVAE) to adjust feature dimensionality and create diverse attack samples. This will help in the imbalance training data problem, which is a challenge for most of the IDS. The author used a trained encoder in the weight initialization of the six-layer deep neural network (DNN) classifier, which could boost the detection performance. However, the model complexity was high because it utilized two DNNs trained by backpropagation. A previous paper proposed an intrusion detection framework known as DDoSNet for detecting DDoS attacks in an SDN-enabled network [20]. The paper proposed a hybrid approach where the auto-encoder is combined with RNN for DDoS threat detection. The developed technique was tested on a new and diverse dataset, CICDDoS2019, and results showed a significant improvement in threat detection compared to traditional ML techniques.

The authors proposed a hybrid deep learning technique based on CNN and LSTM for detecting the DDoS threat [21]. The hybrid deep learning technique is applied to the open-flow SDN controller, which can learn the features of the data and detect the DDoS threat in real time. The simulation results show that the proposed hybrid DNN model can accurately detect the DDoS attack.

Reference [22] added an enhanced history-based IP filtering (eHIPE) stage to the SVM and a self-organizing map (SOM) suggested by Phan et al. to detect attack traffic. The resulting hybrid model was trained and tested using the CAIDA dataset and reached 99.30% accuracy against both ICMP and SYN TCP types of DDoS attacks, thus beating the accuracy of the original SVM-SOM classifier of 97.62%. It also significantly lowered the rate of false alarms from 3.2% to 0.67%. However, for an imbalanced dataset such as CAIDA, the accuracy rate is not a suitable evaluation metric since it does not consider the number of samples in each class. Xu et al. [23] used a voting scheme based on a fast KNN binary classifier (K-FKNN) and "K-means++" for preprocessing to detect attack traffic. The model was trained on the NSL-KDD dataset and achieved an F1 score of 97.65%; it performed far better than other voting algorithms like KD-tree, DPTCM-KNN, and KNN and other classifiers, such as SOM, SVM, and SVM-SOM, and slightly beat probabilistic neural networks. To assess the stability of the results, they tested the model 10 times with test flows ranging from 2000 to 20,000. The proposed method registered the lowest variance of F1 score. K-FKNN significantly increased the detection speed of the voting-based models except for KD-tree.

One of the key issues in existing SDN-based DDoS detection is the constrained communication capacity of the SDN controller that results in reducing the performance of the network. With the aim of addressing the issue, a feature selection technique is proposed where the specific important features are selected from the dataset for detection of the DDoS threat [24]. The simulation is done in an environment where the ML techniques such as NB, SVM, KNN, and ANN classification models are tested without feature selection methods in the datasets, and then the datasets with feature selection are used for training. The results show that the feature selection technique using a KNN technique achieves an accuracy of 98.3% compared to other techniques. Moreover, the feature selection method also reduced the training time of the algorithms. Similarly, the authors of [25] proposed an adaptive polling-based sampling and sFlow technique to reduce the overhead issue of the SDN-enabled network. Then, the deep learning technique based on stacked autoencoders was proposed to detect the DDoS threat. The performance of the proposed framework showed an improved accuracy of 95% for detecting the DDoS threat.

3 Methodology

This section discusses details about the base components of the proposed DDoS detection model, including the algorithms and ensemble techniques used. Moreover, it discusses the dataset used to evaluate our model and the feature selection process.

3.1 Algorithms

Due to the massive amount of network data, the new network architecture of SDN, and various new attacks, traditional classifier algorithms usually perform poorly [19,26]. Considering the above aspects, we propose an intrusion detection method based on DL classifiers.

3.1.1 Convolutional Neural Network (CNN)

CNN is a type of DL model designed to process rich data representations such as images and videos. The main building blocks of a CNN network are the convolutional layers. It consists of a set of the same-size squared matrices called kernels. An element-wise product between each element of the kernel and the input tensor is calculated at each location of the tensor and summed to obtain the output value in the corresponding position of the output tensor. The vector of output tensors corresponding to each kernel of the convolutional layer is called a feature map. Through backpropagation, the CNN can efficiently learn elements of the kernels of the convolutional layers. These convolution operations can run in parallel with the use of GPUs [27]. Essentially, the convolutional layers are parameterized feature extractors. In this manner, CNNs incorporate the feature extraction step into the learning process. A pooling layer is a subsequent layer to the convolutional layers. Its role is to reduce the dimensionality of the feature map. Then, the output is fed to another chain of convolution and pooling layers to build a deep CNN. The output of this stage is fed to a fully connected layer to extract the final output of the network. The modular nature of CNNs allows for the use of the same feature extraction module for multiple tasks with the same or similar input distributions, a scheme commonly known as transfer learning. CNNs have been heavily used in sequence-to-sequence prediction [28], image classification [27], and signal processing with the help of spectrograms [29].

One-dimensional (1D) CNN is an increasingly popular type of CNN that is like the two-dimensional (2D) CNN, but instead of the use of squared matrices as kernels, it uses 1D arrays. This difference works better for tasks that involve the use of one-dimensional input applications such as speech recognition, vibration-based damage detection, and ECG monitoring. 1D CNNs achieve great results without the need for architectures as deep as their 2D counterparts. This adds up to the fact that 1D convolutions are inherently more computationally efficient. The computational advantage of 1D CNNs makes them appropriate for real-time applications and deployment on edge [30].

3.1.2 Long Short-Term Memory (LSTM)

This algorithm is a type of RNN and is very effective and beneficial because it is designed to prevent the problem of long-term dependency. Since the data collected from network traffic are a kind of time series, we believe that LSTM is suitable for our research [31]. Moreover, LSTM is also effective because it focuses on removing complications from the system of DL. With the help of this algorithm, all the dependencies will be resolved, and the programming code will become more optimized and smoother [32].

3.1.3 Gated Recurrent Unit (GRU)

The GRU is a state-of-the-art generation algorithm of RNN, which is somehow similar to the LSTM. This algorithm became popular because it addresses the vanishing gradient problem like LSTM [33]. On the other hand, GRU also has a different limited number of parameters and provides the most effective results in the prediction and in the training network to make it more efficient. Thus, the GRU algorithm is very efficient in prediction with respect to time. However, the accuracy of the estimation of this algorithm is kept in a better range, even with fewer parameters [34].

3.2 Ensemble Method

Ensemble learning has been presented in the existing literature and is superior to single classifier methods in the domain of anomaly detection [13–17,23,35]. Ensemble learning is an approach in which a set of learning models is combined to enhance predictions performance compared to each separate model. The most common strategies applied for ensemble learning are bagging, boosting, and stacking. The voting mechanism collects and combines the predictions from subset models to generate the results. The average of predictions from models is calculated, which is termed average voting. When predictions of each model are summed for a given class and then the class with majority votes is predicted, it is called majority voting. Moreover, to enhance the accuracy of the majority voting technique, each model is assigned with certain weight based on its capability of learning. Afterward, the prediction of the model is multiplied by its model weight before summation. As weights are applied to gain optimality in this approach, such a technique is termed optimally weighted majority voting [36]. In this research, different classifier architectures were used to obtain the ensemble decision, such as 1D-CNN, 2D-CNN, LSTM, and GRU. Using a diversity of models is a prominent factor that yields better results in the final ensemble prediction [37]. Moreover, we applied three different approaches, specifically average voting, majority voting, and optimal weights of individual base learners, to combine the classifiers' predictions for achieving better prediction accuracy.

3.3 Dataset

The rationale behind selecting the CICIDS-2017 dataset was to have an up-to-date and reliable dataset that is considered to be a sufficient benchmark dataset. Flow-based datasets are suitable to evaluate the IDS in work experiments related to the SDN environment in comparison to IP-based traffic. Most of the current network datasets, such as NSL-KDD, KDD99, and DARPA98, are not sufficient for the SDN architecture. The CICIDS-2017 dataset utilizes seven types of up-to-date attacks that are major threats to the SDN. There are seven small datasets; each dataset includes 80 flow features that represent different attack scenarios. For more details about the features, see [38]. The dataset, "Fri-day-Working Hours-Afternoon", was well suited for our investigation on binary classification for DDoS. Tab. 1 provides the attack and benign record numbers.

Table 1: Dataset summary

Traffic type	Number of instances
Benign	128,027
DDoS	97,718

3.4 Dataset Preparation

Building an efficient DL model likely depends on the preprocessing phase. To achieve that, we first needed to remove the redundant data from the dataset [39]. A shuffle method was used to select 140,000 records randomly for classification. A method from the Scikit-learn library was applied to split the dataset as follows: 80% of the records were used in training and 20% for testing. All the selected features for training within each record were normalized into the range of 0–1 with min–max scaling before feeding them to the DL models. See Eq. (1) for the normalization formula.

$$x_{scaled} = \frac{x - \min(x)}{\max(x) - \min(x)} \tag{1}$$

3.5 Features

Restricting the type and number of features affects the detection accuracy and the computation time [39]. In this research, the best appropriate features were selected as recommended in the literature [38]. The "Bwd Packet Length Std" is a complex feature that is significant for detecting DDoS and is not available on NSL-KDD [39]. Tab. 2 shows the description of the selected features.

Table 2: Description of features

Features	Description
Bwd packet length std	Standard deviation size of the packet in the backward direction
Average packet size	The average size of the packet
Flow duration	The total duration of the flow
Flow IAT std	Standard deviation of packets flow inter arrival time

4 The Proposed Ensemble Model

This section introduces the details of our proposed ensemble model construction to select the best model for integration with any commercial SDN controller, as adding the IDS as an application on the control plane is cost-effective and scalable. The process flow of the proposed system and the algorithm are shown in Figs. 2 and 3.

To build the first two types of models, A and B, we used three base learners and applied three methods to combine their predictions: average voting, majority voting, and optimal weights. In the second type, we used two base learners for the ensemble models C and D and applied average voting and optimal weights to predict the final decisions. Tab. 3 lists the construction of four ensemble models. See Figs. 4 and 5 for details.

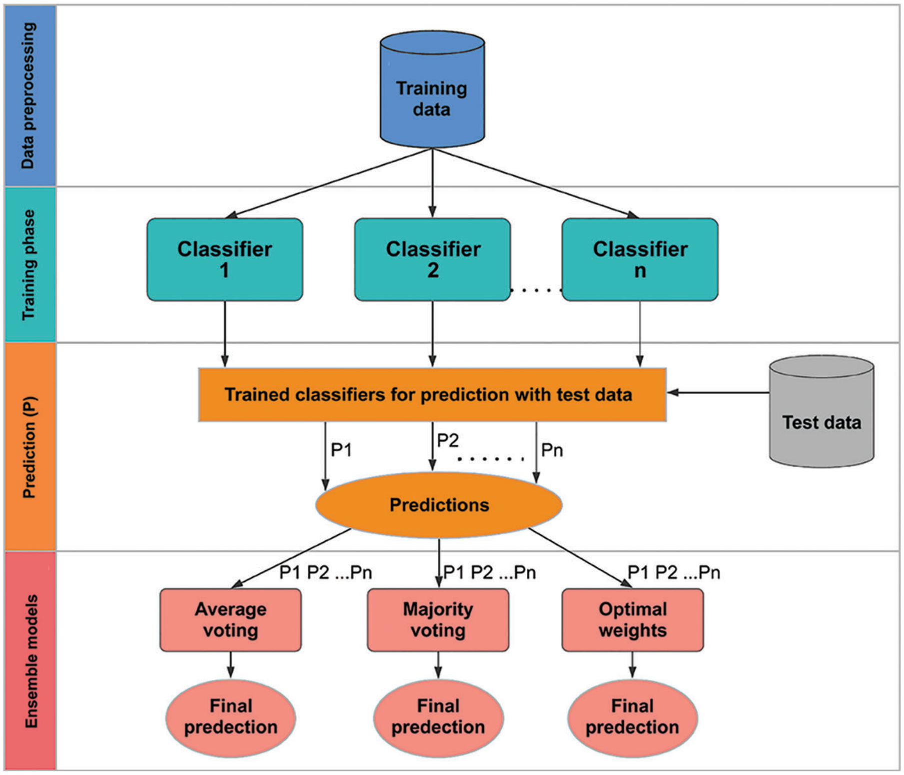
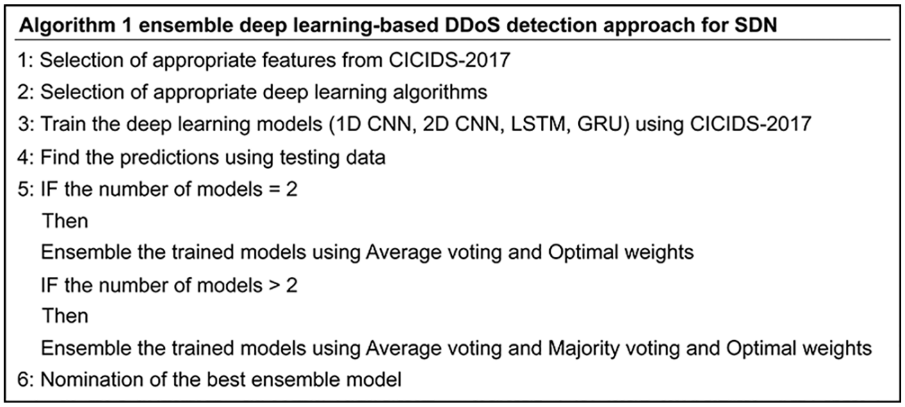


Figure 2: Process flow of the proposed model



Algorithm 1 ensemble deep learning-based DDoS detection approach for SDN

```
1: Selection of appropriate features from CICIDS-2017
2: Selection of appropriate deep learning algorithms
3: Train the deep learning models (1D CNN, 2D CNN, LSTM, GRU) using CICIDS-2017
4: Find the predictions using testing data
5: IF the number of models = 2
   Then
   Ensemble the trained models using Average voting and Optimal weights
   IF the number of models > 2
   Then
   Ensemble the trained models using Average voting and Majority voting and Optimal weights
6: Nomination of the best ensemble model
```

Figure 3: Algorithm for ensemble deep learning-based DDoS detection

Table 3: Construction of ensemble models

Model	Construction
A	1D CNN, 2D CNN, LSTM
B	1D CNN, 2D CNN, GRU
C	1D CNN, 2D CNN
D	LSTM, GRU

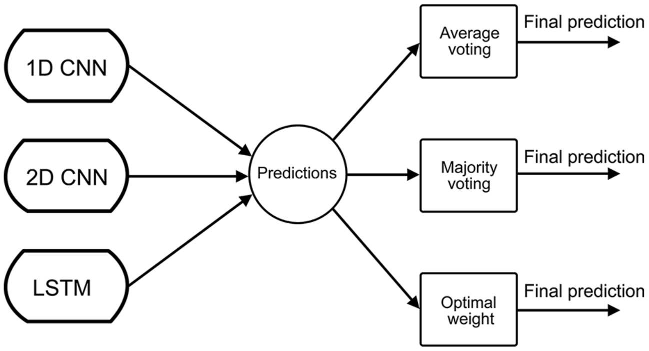


Figure 4: Model A

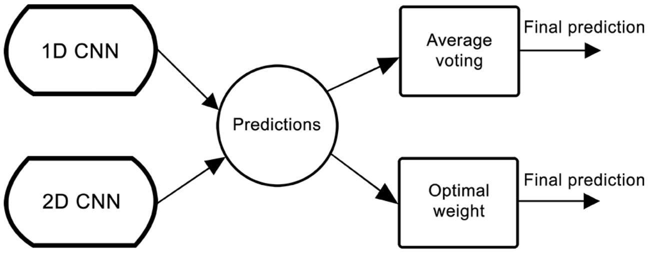


Figure 5: Model C

5 Experiment

5.1 Experimental Setup

The experiment was conducted in Python with the help of several libraries such as Keras, Numpy, Scikit-Learn, and Pandas. We evaluated the best performance by hyperparameter tuning. For designing convolutional layers for CNN models, the filter settings are essential. The number of filters is increased two times in each layer, where the layers close to the input channels have a smaller number of filters (32). The hidden layers were 3 Conv,1 MaxPooling, 1 Flatten, and 2 Dense. RMSProp and Adam algorithms are considered effective optimizers for deep neural networks [40]. RMSProp was selected for the best results. All experiments were performed for the selection of activation functions, showing that the smooth approximation of the rectified linear unit (ReLU) function achieved the best performance. The number of epochs was equal to 20 for models A, B, C, and D. Based on each DL model, dimensional reconstruction was used to fit the model input requirements. We also applied a single model for comparison with ensemble models. See Tab. 4 for the parameters used during the experimentation.

Table 4: Hyperparameter values

Activation function	Loss function	Batch size	Optimization algorithm	Input layer	Number of epochs
CNN: rectified linear unit (ReLU) for all hidden layers, sigmoid for output layers LSTM/GRU: sigmoid	Cross-Entropy Loss	128	RMSProp	4	20

5.2 Results and Discussion

In this section, the anomaly detection performances of our models A, B, C, and D are demonstrated using accuracy, precision score, recall score, and F1 score. Details of the results are given in Tabs. 5 and 6 and Figs. 6–8, which show that our models perform well for all the evaluation metrics. We compared the performance of our best-performing model with those of others in the literature. For further investigation, we also compared the performance of our proposed models with those of single models such as 1D CNN, 2D CNN, LSTM, and GRU, using the same subset of four features.

Table 5: Test accuracy of the proposed models

Model	Average voting	Majority voting	Optimal weights
A	99.7	99.55	99.63
B	99. 69	99.58	99.66
C	99.77	N/A	99.73
D	98.73	N/A	98.71

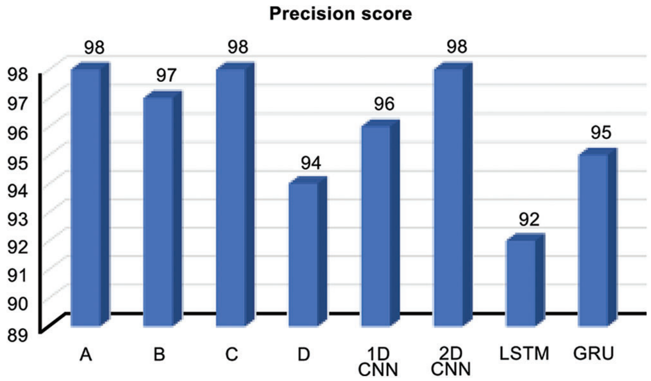


Figure 6: Precision score for each model

5.2.1 Accuracy

In our experiments, the accuracy was very high, and for all the models, it exceeded 98%. This means that the DL learning model correctly classified almost 98% of all the observations in the test data. The selection of the most significant features affects the accuracy. The method of average voting was superior to the majority voting and optimal weights. The best accuracy (99.77%) in this experiment was achieved by model C, which was constructed by combining two different architectures of CNN using the average voting method. Model D resulted in the lowest accuracy (98.71%). Additionally, we observed that the ensemble models beat the single models in terms of accuracy (see Tabs. 5 and 6).

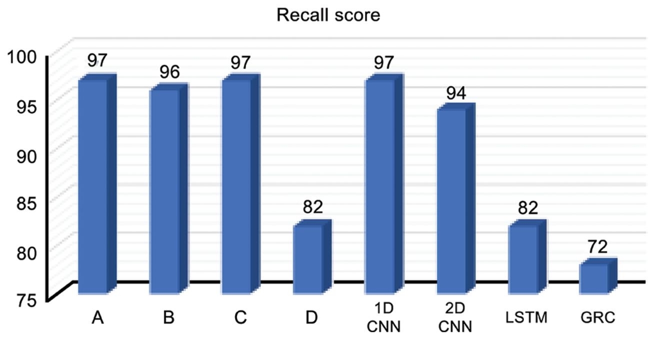


Figure 7: Recall score for each model

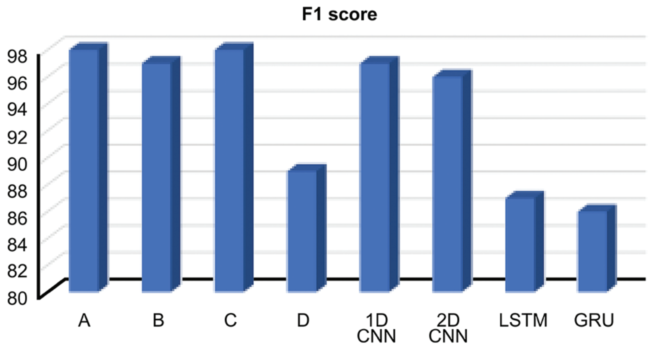


Figure 8: F1 Score for each model

Table 6: Single model accuracies

Model	Accuracy
1D CNN	99.64
2D CNN	99.59
LSTM	98.70
GRU	98.59

5.2.2 Precision Score

The precision score measures the classifiers' exactness and is calculated as follows:

$$\frac{TP}{TP + FP}$$

High precision signals the presence of a low false-positive rate. In this experiment, the precision score of the predictions was around 98% for models A and C. This means that of all the normal incidences predicted by the model, 98% of them were correctly classified, while the remaining 2% were attack events wrongly classified as normal events. For comparing the precision score values, we selected the results from the ensemble model using average voting because it obtained the best results. From Fig. 6, ensemble models including CNN achieved better results. The single model GRU achieved a better value than model D, and the 2D-CNN model obtained an equal value with models A and C.

5.2.3 Recall Score

This is a measure of the model's correctness and is calculated as follows:

$TP/(TP + FN)$

The recall score was 97% for models A and C. This signifies that out of all normal incidences in the dataset, 97% were correctly predicted as normal incidences by the model. The recall is not as high as the accuracy measure because the model has also learned certain patterns of attack labeled rows; hence, it results in fewer TP. The 1D-CNN model achieved the same value as models A and C (see Fig. 7).

5.2.4 F1 Score

The F1 score is the weighted average of precision and recall. Since the F1 score considers both the FP and the FN, it is a useful measure when evaluating a prediction model. The highest F1 score was 98% for models A and C, as shown in Fig. 8. This implies that there is a good balance between precision and recall scores. The ensemble models acquired better results compared to single models, except for 1D CNN, which achieved the same result as model B. Model C, which was constructed by averaging the votes of 1D CNN and 2D CNN, yielded the best results. Model A, which came in second place, gained better results than model B; this implies that the LSTM model has a better effect than the GRU model. The CNN models in the ensemble and single models produced better results.

As can be seen from the results above, model C gives us the best results among all the other three proposed DL approaches and single models, as per almost all the evaluation metrics. For further evaluation, we concluded other measures for the model C performance. Fig. 9 shows the false-positive rate (FPR), false-negative rate (FNR), false omission rate (FOR), and false discovery rate. FPR is an important parameter since the cost of false alerts is high in SDN, where the mitigation policies create extra work. The performance comparison among all the ensembles model in terms of FBR and FOR is demonstrated in Fig. 10. The area under the curve and receiver-operating characteristics (ROC) is a commonly used metric and demonstrated for model C in Fig. 11.

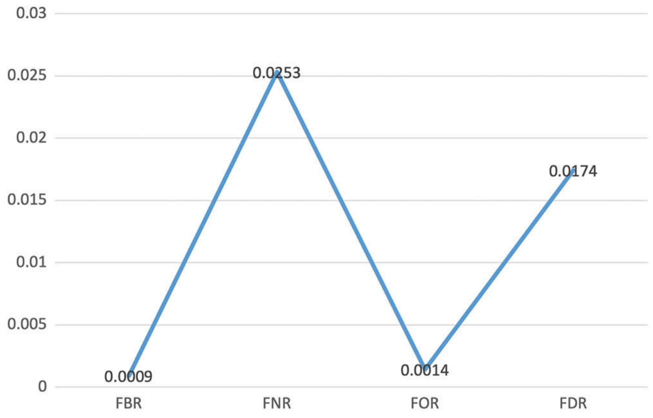


Figure 9: Performance of model C

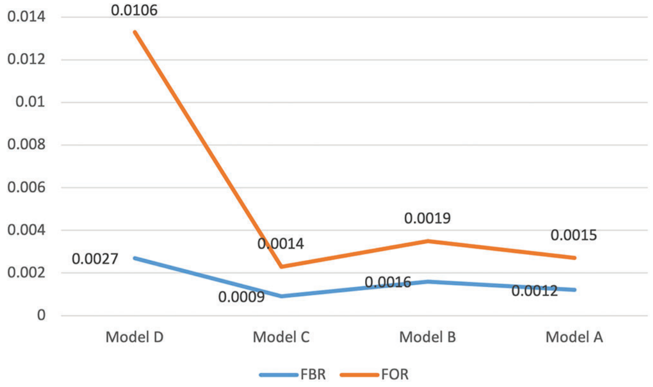


Figure 10: Comparison of ensemble models using false-positive rate and false omission rate

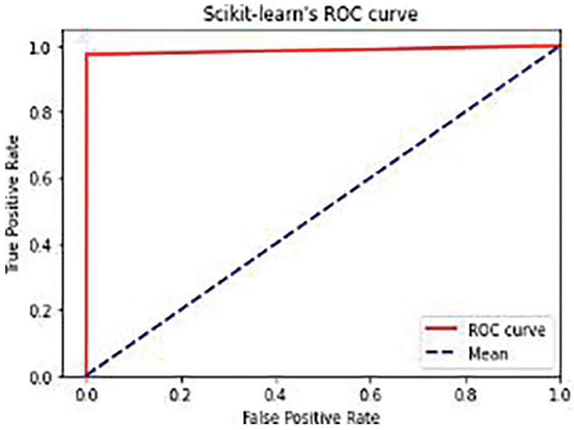


Figure 11: Receiver-operating characteristics (ROC)

5.2.5 Comparison with Previous Studies

The accuracy comparison of model C with other competing approaches in the literature is demonstrated in Tabs. 7 and 8. Tab. 7 shows that our proposed model achieves better accuracy than all the previous methods that use the same dataset for evaluation. Our ensemble CNN achieved better accuracy than the ensemble CNN in [13] and [14], which used two similar CNN models. Compared to [13] and [14], we increased the accuracy to 99.77% and reduced the FPR to 0.0009 for the same data dataset and number of features. Tab. 8 compares our proposed model with existing approaches presented in [15,17,19,20,22], which used other datasets such as CAIDA, NSL-KDD, and UNSW.

Table 7: Accuracy comparison with previous studies on DDoS detection using the CICIDS2017 dataset

Method	Accuracy	Number of features
Model C (Proposed Model)	99.77 %	4
[14] Ensemble CNN	99.48%	4
[13] Ensemble CNN	99.45%	4
[18] (SAE-1SVM)	99.35%	13
[11] GRU-RNN	99%	9

Table 8: Accuracy comparison with previous studies on DDoS detection with deferent datasets

Method	Accuracy	Dataset
Model C (Proposed model)	99.77%	CICIDS207
[22] SVM+ SOM	99.30%	CAIDA
[15] Ensemble Model	99.1%	NSL-KDD
[17] Ensemble of ELMs	98.24%	UNSW
[20] RNN-Autoencoder	99%	CICDDoS2019
[19] ICVAE-DNN	85.97%	NSL-KDD

6 Conclusion and Future Research

SDNs require an efficient approach in the prevention and detection of security attacks. The work from the past few years has shown the implementation and success of ensemble learning. Ensemble DL-based approaches show their strong ability to work with diverse and complex traffic for intrusion detection in the SDN paradigm. The DDoS detection accuracy must be improved to ensure resiliency in SDN and service availability. The proposed ensemble models demonstrate promising results in anomaly detection and can work well with several types of DDoS in the CICIDS2017 dataset. An accuracy of 99.77% was achieved by the proposed ensemble CNN solution, which constructs two different architectures of CNN and only four features. Based on the results from the conducted experiments, the proposed ensemble CNN was able to detect the DDoS attack with higher accuracy compared with other proposed ensemble models and single classifiers. This implies that CNN-based ensembles have a promising potential to offer better IDS solutions for SDN and need further research. We evaluated our proposed solutions with a benchmark flow-based SDN dataset. However, it would be better to subject this solution to additional evaluation in the SDN testbed with real traffic. The training process can be investigated for more optimization to accomplish better results. To further refine the test results, we can employ advanced ensemble techniques including stacking.

Acknowledgement: The authors would like to thank the Deanship of Scientific Research (DSR), King Abdulaziz University, Jeddah, Saudi Arabia, for technical and financial support.

Funding Statement: This project was funded by the Deanship of Scientific Research (DSR) at King Abdulaziz University, Jeddah, under grant no. (RG-8-611-40).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] N. Anerousis, P. Chemouil, A. A. Lazar, N. Mihai and S. B. Weinstein, "The origin and evolution of open programmable networks and SDN," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 3, pp. 1956–1971, 2021.

[2] N. Faujdar, A. Sinha, H. Sharma and E. Verma, "Network security in software defined networks (SDN)," in *Proc. Int. Conf. on Smart Technologies in Computing, Electrical and Electronics (ICSTCEE) 2020*, Bengaluru, India, pp. 377–380, 2020.

[3] S. Dong, K. Abbas and R. Jain, "A survey on distributed denial of service (DDoS) attacks in SDN and cloud computing environments," *IEEE Access*, vol. 7, pp. 80813–80828, 2019.

[4] “Aws-shield-tlr.s3.amazonaws.com”, [Online]. Available: https://aws-shield-tlr.s3.amazonaws.com/2020-Q1_AWS_Shield_TLR.pdf%0A (Accessed: June 03, 2021).

[5] T. A. Tang, L. Mhamdi, D. McLernon, S. A. R. Zaidi, M. Ghogho *et al.,* “DeepIDS: Deep learning approach for intrusion detection in software defined networking,” *Electronics*, vol. 9, no. 9, pp. 1533, 2020.

[6] O. Rahman, M. A. G. Quraishi and C. H. Lung, “DDoS attacks detection and mitigation in SDN using machine learning,” *2019 Proc. of IEEE World Congress on Services*, Milan, Italy, vol. 2642-939X, pp. 184–189, 2019

[7] A. Patil, P. Jain, R. Ram, V. Vayachal and S. Bendale, “Detection of distributed denial-of-service (DDoS) attack on software defined network (SDN),” *International Research Journal of Engineering and Technology*, vol. 918, pp. 918–921, 2018.

[8] N. Sultana, N. Chilamkurti, W. Peng and R. Alhadad, “Survey on SDN based network intrusion detection system using machine learning approaches,” *Peer-to-Peer Networking and Applications*, vol. 12, no. 2, pp. 493–501, 2019.

[9] N. Ravi and S. M. Shalinie, “Learning-driven detection and mitigation of DDoS attack in IoT via SDN-cloud architecture,” *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 3559–3570, 2020.

[10] S. S. Mohammed, R. Hussain, O. Senko, B. Bimaganbetov, J. Lee *et al.,* “A new machine learning-based collaborative DDoS mitigation mechanism in software-defined network,” in *2018 14th Int. Conf. on Wireless and Mobile Computing, Networking and Communications (WiMob)*, Limassol, Cyprus, pp. 1–8, 2018.

[11] T. A. Tang, D. McLernon, L. Mhamdi, S. A. R. Zaidi and M. Ghogho, “Intrusion detection in SDN-based networks: Deep recurrent neural network approach,” in *Deep Learning Applications for Cyber Security (Advanced Sciences and Technologies for Security Applications)*, 1st. edition, Berlin/Heidelberg, Germany, Springer International Publishing, pp. 175–195, 2019.

[12] S. K. Dey and M. M. Rahman, “Flow based anomaly detection in software defined networking: A deep learning approach with feature selection method,” in *Proc. 4th Int. Conf. on Electrical Engineering and Information & Communication Technology (iCEEiCT)*, Bangladesh, pp. 630–635, 2018.

[13] S. Haider, A. Akhunzada, I. Mustafa, T. B. Patel, A. Fernandez *et al.,* “A deep CNN ensemble framework for efficient DDoS attack detection in software defined networks,” *IEEE Access*, vol. 8, pp. 53972–53983, 2020.

[14] S. Haider, A. Akhunzada, G. Ahmed and M. Raza, “Deep learning based ensemble convolutional neural network solution for distributed denial of service detection in SDNs,” in *2019 UK/China Emerging Technologies (UCET)*. Glasgow, Scotland, UK, 1–4, 2019.

[15] S. Das, A. M. Mahfouz, D. Venugopal and S. Shiva, “DDoS intrusion detection through machine learning ensemble,” in *2019 IEEE 19th Int. Conf. on Software Quality, Reliability and Security Companion (QRS-C)*, Sofia, Bulgaria, pp. 471–477, 2019.

[16] R. Swami, M. Dave and V. Ranga, “Voting-based intrusion detection framework for securing software-defined networks,” *Concurrency and Computation: Practice and Experience*, vol. 32, no. 24, pp. e5927, 2020.

[17] J. Sharma, C. Giri, O. C. Granmo and M. Goodwin, “Multi-layer intrusion detection system with ExtraTrees feature selection, extreme learning machine ensemble, and softmax aggregation,” *EURASIP Journal on Information Security*, vol. 2019, no. 1, pp. 1–6, 2019.

[18] L. Mhamdi, D. Mclernon, F. El-Moussa, S. A. Zaidi, M. Ghogho *et al.,* “A deep learning approach combining autoencoder with one-class SVM for DDoS attack detection in SDNs,” in *2020 IEEE 8th Int. Conf. on Communications and Networking (ComNet)*, Hammamet, Tunisia, 2020.

[19] Y. Yang, K. Zheng, C. Wu and Y. Yang, “Improving the classification effectiveness of intrusion detection by using improved conditional variational autoencoder and deep neural network,” *Sensors (Switzerland)*, vol. 19, no. 11, pp. 2528, 2019.

[20] M. S. Elsayed, N. A. Le-Khac, S. Dev and A. D. Jurcut, “Ddosnet: A deep-learning model for detecting network attacks,” in *2020 IEEE 21st Int. Sym. on A World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, Cork, Ireland, pp. 391–396, 2020.

[21] C. Li, Y. Wu, X. Yuan, Z. Sun, W. Wang *et al.,* “Detection and defense of DDoS attack-based on deep learning in OpenFlow-based SDN,” *International Journal of Communication Systems*, vol. 31, no. 5, pp. e3497, 2018.

[22] T. V. Phan and M. Park, “Efficient distributed denial-of-service attack defense in SDN-based cloud,” *IEEE Access*, vol. 7, pp. 18701–18714, 2019.

[23] Y. Xu, H. Sun, F. Xiang and Z. Sun, “Efficient DDoS detection based on K-FKNN in software defined networks,” *IEEE Access*, vol. 7, pp. 160536–160545, 2019.

[24] H. Polat, O. Polat and A. Cetin, “Detecting DDoS attacks in software-defined networks through feature selection methods and machine learning models,” *Sustainability*, vol. 12, no. 3, pp. 1035, 2020.

[25] R. M. A. Ujjan, Z. Pervez, K. Dahal, A. K. Bashir, R. Mumtaz *et al.,* “Towards sFlow and adaptive polling sampling for deep learning based DDoS detection in SDN,” *Future Generation Computer Systems*, vol. 111, no. 10, pp. 763–779, 2020.

[26] P. Ding, J. Li, L. Wang, M. Wen and Y. Guan, “HYBRID-CNN: An efficient scheme for abnormal flow detection in the SDN-based smart grid,” *Security and Communication Networks*, vol. 2020, no. 4, pp. 1–20, 2020.

[27] W. Rawat and Z. Wang, “Deep convolutional neural networks for image classification: A comprehensive review,” *Neural Computation*, vol. 29, no. 9, pp. 2352–2449, 2017.

[28] M. Elbayad, L. Besacier and J. Verbeek, “Pervasive attention: 2D convolutional neural networks for sequence-to-sequence prediction,” in *Proc. 22nd Conf. on Computational Natural Language Learning, CoNLL 2018*, Brussels, Belgium, pp. 97–107, 2018.

[29] S. Kwon, “A CNN-assisted enhanced audio signal processing for speech emotion recognition,” *Sensors (Switzerland)*, vol. 20, no. 1, pp. 183, 2020.

[30] S. Kiranyaz, O. Avci, O. Abdeljaber, T. Ince, M. Gabbouj *et al.,* “1D convolutional neural networks and applications: A survey,” *Mechanical Systems and Signal Processing*, vol. 151, pp. 107398, 2021.

[31] R. Madan and P. Sarathimangipudi, “Predicting computer network traffic: A time series forecasting approach using DWT, ARIMA and RNN,” in *11th Int. Conf. on Contemporary Computing, IC3 2018*, Noida, India, pp. 2–4, 2018.

[32] Z. Sun, L. Di and H. Fang, “Using long short-term memory recurrent neural network in land cover classification on Landsat and Cropland data layer time series,” *International Journal of Remote Sensing*, vol. 40, no. 2, pp. 593–614, 2019.

[33] J. Yang, L. Zhang, C. Chen, Y. Li, R. Li *et al.,* “A hierarchical deep convolutional neural network and gated recurrent unit framework for structural damage detection,” *Information Sciences (NY)*, vol. 540, no. 1, pp. 117–130, 2020.

[34] H. Fanta, Z. Shao and L. Ma, “SiTGRU: Single-tunnelled gated recurrent unit for abnormality detection,” *Information Sciences*, vol. 524, pp. 15–32, 2020.

[35] V. Deepa, K. M. Sudar and P. Deepalakshmi, “Design of ensemble learning methods for DDoS detection in SDN environment,” in *ViTECoN 2019, Proc.: Int. Conf. on Vision Towards Emerging Trends in Communication and Networking*, Vellore, India, pp. 1–6, 2019.

[36] J. Kazmaier and J. H. van Vuuren, “The power of ensemble learning in sentiment analysis,” *Expert Systems with Applications*, vol. 187, no. 2, pp. 115819, 2022.

[37] T. Pang, K. Xu, C. Du, N. Chen and J. Zhu, “Improving adversarial robustness via promoting ensemble diversity,” in *Int. Conf. on Machine Learning*, Long Beach, California, USA, pp. 4970–4979, 2019.

[38] I. Sharafaldin, A. H. Lashkari and A. A. Ghorbani, “Toward generating a new intrusion detection dataset and intrusion traffic characterization,” in *ICISSP 2018. Proc. 4th Int. Conf. on Information Systems Security and Privacy*, Funchal-Madeira, Portugal, pp. 108–116, 2018.

[39] D. Stiawan, M. Y. Bin Idris, A. M. Bamhdi and R. Budiarto, “CICIDS-2017 dataset feature analysis with information gain for anomaly detection,” *IEEE Access*, vol. 8, pp. 132911–132921, 2020.

[40] R. S. Srinivasamurthy, “Understanding 1D convolutional neural networks using multiclass time-varying signals,” ProQuest Dissertations and Thesis, 99, 2018.