Adapting Deep Learning-LSTM method in SDN controller For Secure IoT

Omer Elsier (oalser@kku.edu.sa)
King Khalid University
AZATH MUBARAKALI
King Khalid University
Amira Elsir Tayfour
King Khalid University
Muhammad Nadzir Marsono
Universiti Teknikal Malaysia - Main Campus: Universiti Teknikal Malaysia Melaka
Entisar Hassan
Universiti Teknikal Malaysia - Main Campus: Universiti Teknikal Malaysia Melaka
Ashraf M. Abdelrahman
Integrated Thebes Institutes: Thebes Academy

Research Article

Keywords:

Posted Date: March 1st, 2023

DOI: https://doi.org/10.21203/rs.3.rs-2596275/v1

Version of Record: A version of this preprint was published at Soft Computing on May 9th, 2023. See the published version at https://doi.org/10.1007/s00500-023-08348-w.

Abstract

The Internet of Things (IoT) has grown into various enterprise. While the IoT ecosystem's extensive and open environment has many advantages, it can also be a target for a range of growing cyber risks and assaults. The benefits of device integration into a smart ecosystem are enhanced by the IoT's diversity, but the IoT's diverse nature makes establishing a single security solution difficult. However, software-defined networks' (SDNs) centralized intelligence and programmability, it's now possible to put together a single, effective security solution to combat cyber threats and attacks. This study proposes a DL-driven SDN-enabled IoT framework that practice a Deep Neural Network-Long Short-Term Memory (LSTM) classifier to quickly and efficiently detect sophisticated multisector malware botnets. The proposed mechanism was rigorously tested utilizing the most recent state-of-the-art dataset, CICIDS2017, as well as traditional performance evaluation metrics. Furthermore, the proposed technique is compared to current industry norms (i.e., DL algorithms). Extensive testing shows that the proposed method surpasses the competition in terms of detection accuracy while requiring just a minimal compromise in terms of computational cost.

Introduction

Security has become increasingly crucial as the Web has grown and each networking device has been interconnected. The Internet of Things (IoT) is causing a slew of security issues. IoT is a worldwide network of networked gadgets with unique addresses. Sensors and numerous communication protocols are used by IoT devices. They have both the computational and data-evaluation abilities to give services. Cameras, IP cameras, all types of sensors are examples of IoT gadgets. IoT is a rapidly evolving technological environment. It's a pattern that connects millions of smart gadgets to form a smart environment [1].

Many obstacles must be addressed before tangible implementations for such complicated environments can be provided, with security and confidentiality being serious supports [2], Especially in the context of safety applications, when key decisions are reliant on information gathered by users regarding their condition or surrounding occurrences [3]. The Internet of Things is made up of disparate networks and connecting devices that communicate through number of protocols. The active properties of IoT devices raise a number of security problems in the form of denial of service (DoS) attacks, distributed denial of service (DDoS) assaults, and other malware [4–6]. Cyber-physical systems have advanced rapidly in recent years as computing and hardware technology have advanced. However, as these sectors have improved and developed, so has the range of cyber-attacks; for example, DoS/DDoS attacks, which render the system's resources inaccessible. [7] and [8] examine attack detection methodologies at the engineering level. Traditional intrusion detection solutions defend devices against assaults by deploying firewalls or intrusion detection and prevention systems at the infrastructure level. However, due to the seamless nature of IoT devices, these security procedures are inadequate. Network security features include firewalls, antivirus software, and intrusion detection systems (IDSs). An intrusion detection system (IDS) identifies unauthorized system behavior.

In this context, attackers can infiltrate networking devices and network sensitive information using a range of exploits such as phishing, DoS, DDoS, a variability of malware assaults, and Botnets in an attempt to cooperation the whole system's operation. Botnet, for example, is one of the most dangerous sorts of assault, accomplished of incapacitating the complete network. A Botnet is a collection of hacked connected devices that are controlled remotely with the intention of committing online crimes by exploiting vulnerabilities [9]. By transmitting instructions to the stolen systems, an attacker can use this channel to steal data from users [10]. One of the greatest strategies for software defined network surveillance [11] is to adapt IDS in an SDN to address these important security, trust, and operational assurance concerns.ML-based techniques for various detection processes in SDNs have been documented in the literature [12–13]. On the other hand, there is a trend and a modest movement from ML to DL-based techniques in the present literature [14], [15], [16]. The reason for this is because DL is incredibly effective and does not require any additional processing for feature selection. Furthermore, the capacity to detect zero-day assaults is a benefit of DL-based anomaly detection systems.

Key of Contributions:

- This research, focus an SDN-based Deep Learning framework that uses LSTM classification models to identify the threats in IoT environments early and efficiently.
- For a multi-class attack detection system, IoT flow-based dataset (i.e., CICIDS2017) was used to test and evaluate the newly built architecture.
- We compare the proposed system to current benchmark algorithms for a comprehensive evaluation utilizing standard and extended evaluation measures to determine the system's true performance (i.e., accuracy, precision, recall, and F1-score).

The balance of the research is placed out as follows. The literature is discussed in Second Section. The proposed model, as well as the methods, are described in Section III. In Section IV, the results and analysis are presented. Finally, Section V comes to a close.

1.1 RELATED WORK

In a variety of computer science disciplines, deep learning has led to tremendous advances. The authors presented a DL based IDS in software defined networks using the Restricted Boltzmann Machine (RBM) in [17]. For experimental testing, the technique used the CMU-based assaults dataset and the KDD99 dataset, achieving a detection accuracy of 99.98 percent for binary classification.

Tang et al. [18] introduced a deep Gated Recurrent Unit Recurrent Neural Network (GRU-RNN) based IDS based on NSL-KDD and CICIDS2017 dataset, which reached 89 percent multiclass detection accuracy. The design of Software Design Network Included on Flows and NSL-KDD, on the other hand, is not at all flow-based. In [19], the writers offer a deep learning system for risk and attack detection in SDN based on Multi-Layer Perceptron (MLP). Using the CTU-13 dataset, the suggested framework achieves a detection accuracy of 98.7%.

Deep learning (ANN, CNN, and LSTM)-based cybersecurity architecture was proposed in [20] to identify vulnerabilities in IoT networks. The LSTM model achieved an accuracy of 87 percent. Furthermore, in [21], the writers proposed a network intrusion detection framework based on Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM), and Convolutional Neural Networks (CNN).Utilizing six essential properties of the NSL-KDD dataset, developed a network IDS using Gated Recurrent Unit Recurrent Neural Network (GRU-RNN) in [22]. The proposed methodology achieves an accuracy of 89 percent in detecting cyber threats and assaults, which is insufficient to identify real-time changing cyber-attacks. Author [23] has proposed to hybrid- GRU-LSTM to identify the Flow-based anomaly detection in SDN. Feature Selection methods is used to improve detection accuracy up to 87% as compared with NSL-KDD Dataset.

Recurrent neural network (RNN) techniques were utilized by the authors in [24] to recognize and categorize attacks. The performance of RNN-based techniques and non-RNN techniques was compared. IDS model was suggested, constructed, and trained using single CNN models and a multi-CNN fusion model in this [25]. The NSL-KDD was used to train these models. The multi-CNN fusion model achieved extraordinary accuracy of 86.95 percent when compared to well-known machine learning classification methods and the latest deep learning algorithms. In [26], the authors described an SDN-assisted hybrid deep learning technique for detecting bots in the Internet of Medical Things.Despite the good identification accuracy achieved, the data from the literature, which are also shown in Table 1, suggest that there is still space for improvement.

Table 1
Assessment of related work

Ref	Attack Type	Dataset	Classifier	Strengths	Weaknesses
[25] 2020	DoS, Probe, U2L,R2L	NSL-KDD	Multi-CNN	Low complexity	Detection accuracy need to improved ,NSL-KDD is not flow based dataset
[17] 2019	DDoS	CMU based insider threat, KDD99	RBM, SVM	Ensemble Classifier with high detection for multi-class	Old, static data
[18] 2019	DoS, Probe, U2L,R2L	NSL-KDD and CICIDS2017	GRU-RNN	Achieved a detection accuracy of 89% for multiclass	Accuracy need to Improve
[19] 2019	Botnet	CTU-13 ISOT	MLP	IDS - MLP to identify botnet	Not performs experiment on terminals that are infected with bot
[20] 2019	Bots	iperf3	ANN, CNN, LSTM	Capability to capture traffic measurements (real time)	Real dataset is not used to prove the work
[21] 2018	DDoS	ISCX2012	RNN, LSTM,CNN	Feature extraction has performed and achieved accuracy 98%	Detection of computational complexity and time overhead
[22] 2018	DoS, U2L,R2L	NSL-KDD	GRU-RNN	Achieved a detection accuracy of 89% using just 6 network features	The detection accuracy is insufficient to detect intrusions in real time.
[23] 2018	DoS, Probe	NSL-KDD	GRU-LSTM	Hybrid framework	NSL-KDD is not a dataset that is based on flow.
[24] 2017	DoS, Probe, U2L,R2L	KDD99, UNSW-NB15	RNN	High detection rate	Old, static data

Methodology

Despite the good identification accuracy achieved, the data from the literature, which are also revealed in Table 1, suggest that there is still space for improvement.

A. Proposed Framework

Figure 1 depicts the overview of the SD-IoT architecture, as well as our suggested control plane structure. The control plane, which is accountable for the entire centralized control intelligence and network management, is where SD-power IoT's rests.

The design architecture of multiple SDN controllers is usually the same, but the functionality vary. Furthermore, the implementation language varies from controller to controller.

B. LSTM Classifier

The proposed deep learning architecture uses long short-term memory (LSTM) to detect the threats, as demonstrated in (Figure.2)

The LSTM-based classification models utilized in our studies are briefly explained in this section.

LSTM is a Recurrent Neural Network enlargement or variant (RNN). Traditional RNNs suffer from short-term memory problems, which means that if the sequence is long, the classifier has trouble carrying information from earlier to later time steps. As a result, the gradient vanishing problem affects recurrent neural networks, and LSTM was created as remedy. The gating system is made up of three gates (i.e., input, output, and forget gate). These gates are in charge of determining whether data sequences should be saved or discarded.

The processes of LSTM are showed in Fig. 3. In general, the LSTM cell's most significant role is to determine whether information should be removed or maintained. Information from a previous hidden state $(\mathrm{h_t}-1)$ and the current input (Xt) are handled by the forget gate. It adds them together and uses the function to generate numbers among 0 and 1. If the value is near to 0, the information is discarded; then, the value is transferred to the next phase, which will analyse the previous cell state $(\mathrm{c_t}-1)$. In addition, the result of $(\mathrm{h_t}-1 + \mathrm{x_t})$ is passed through another σ and tanh function. The outcome would then be multiplied. The following equation is used to calculate the current state $(\mathrm{c_t})$:

$c_t = c_t-1 \otimes f_t \oplus I_t$ (1)

where, $f_t$is the forget gate

$f_t = \sigma(W_f\ (x_t + h_t-1))$ (2)

and *It* is the input gate, which consists of two equations:

$i_{t1} = \sigma\ (W_{i2}\ (x_t + h_t-1))$ (3)

$i_{t2} = \tanh(W_{i1}\ (x_t + h_t-1))$ (4)

With the earlier two equations, we get the input gate I_t value as

$I_t = i_{t1} \otimes i_{t2}$ (5)

In our suggested framework, Table 2 lists the setup parameters of LSTM classifier, including layers, optimizer, batch size, epochs, and activation function.

Table 2
LSTM model configuration

Algorithm Family	Layer	Activation function	Optimizer	Batch-size	Epochs
RNN	LSTM(1) Output Layer (1)	Relu, Softmax	Adam	32	10

C. Dataset

Real-world datasets were used to assess our approach. Our suggested solution is based on CICIDS2017 [26], a flow-based dataset for SDN. Port Scan, Cross Site Scripting (XSS), Brute Force FTP, Brute Force SSH, DoS, Heartbleed, Web Attack, Botnet, and DDoS are among the benign and most recent prevalent assaults in the dataset (Fig. 4). We are concerned with two different application-level assaults (Botnet, Port Scan, and DDoS) and one benign class in this study. Table 3 shows the complete distribution of the CICIDS2017 set used in the suggested technique. Our decision to employ specific classifications (Botnet, Port Scan, and DDoS) arises from a desire to emphasison the discovery of attacks related to reconnaissance, which is essentially a data collection and surveillance technique used to exploit systems. For our experiment, which was based on [28], we only used ten features.

Table 3
CICIDS2017 data for our experiment

Class	Number of Instance
Benign	414322
Bot	1966
Port Scan	158930
DDoS	128037

D. Pre-Processing

Preprocessing of data, which includes managing the absent data, grouping the data, and feature selection of datasets, is the first phase of the research approach (Fig. 2).

E. Experimental Setup

The research has created a framework utilizing the deeplearnig4j library [29] to achieve the best performance out of the suggested intrusion detection architecture for SD-IoT. The practical implementation is carried out in an Intel Core i7 setting. As an alternative of using the k-fold cross-validation method, we employed the train-test split technique to evaluate the model in this study.

F. Evaluation Metrics

The system of measurement utilized in this research to validate the performance of the DDoS attack detection were introduced in this sub-section, and they were expressed as below:

Table 4
Evaluation Metrics

Accuracy	Recall	Precision	F1-Score
$\frac{TP+TN}{TP+FP+TN+FN}$	$\frac{TP}{TP+FN}$	$\frac{TP}{TP+FP}$	$\frac{2*Recall*Precision}{Recall+Precision}$

- True positive (TP): Categorizing abnormal
- False positive (FP): Imperfectly categorizing normal data as an abnormal.
- True negative (TN): Correctly categorizing normal data
- False negative (FN): Imperfectly categorizing attacks as normal.

Results And Discussion

This study adds to the body of knowledge by demonstrating the utility of DL in anomaly detection systems. We presented a DL technique based on LSTM to classify input traffic into regular and hazardous categories. Because of their ability to deal with a high degree of complicated nonlinear interactions, DL approaches appear promise for detecting network intrusion. It can be used to find traffic abnormalities beyond the limitations of typical classification approaches, which rely on domain expertise.

The goal of this research is to see how well our proposed LSTM performs. This section investigates and details the relationship between the conclusions produced using the traditional and extended evaluation metrics discussed before. A thorough comparison of the proposed model with current benchmarks is shown in Table 7.

Table 5 shows the accuracy, precision, recall, and F1-score. Our implemented solution obviously outperforms in terms of key performance measures, as evidenced by this graph. Figure 5, on the other hand, clearly illustrates the average per-class accuracy of various attacks, as well as the fact that LSTM correctly detected three distinct types of attacks (Botnet, Port Scan, and DDoS).

Table 5
Average performance of Evaluation metrics

Multi classes	Parameter				
	Accuracy	Precision	Recall	F1-score	ROC Area
Weighted Average	0.9932	0.993	0.993	0.993	0.999

The Receiver Operating Characteristic (ROC) curve is used to assess how accurate the model is. The relationship among two parameters, True and False classes, is represented by the ROC curve. The AUC

(area beneath the ROC Curve) estimates the rate of false positives and true positives. Figure 6 shows that our model has an AUC of 99.9%, indicating that our suggested model can correctly differentiate 99.9% of positive and negative classes.

In order to describe our model's classification performance, we also show the confusion matrix. The right and false predictions are summarized in the confusion matrix. Table 6 shows that our model has a 0.993 accuracy in detecting all attack and benign classifications.

Table 6
Confusion matrix (TP, FP, TN, FN) for LSTM.

Attack	**0.993**	**0.007**
Benign	0.007	0.993
	Attack	Benign

Table 7
Comparison with benchmarks

Schemes	Algorithms	Datasets	Accuracy%	Precision%	Recall%	F1-score%
Proposed	LSTM	CICIDS2017	99.32	99.3	99.3	99.3
[25]	Multi-CNN	NSL-KDD	86.95	89.56	87.25	88.41
[18]	GRU-RNN	CICIDS2017	89	99	99	99
[23]	GRU-LSTM	NSL-KDD	87.9	83.5	77.9	80.60
[30]	BLSTM	CICIDS2017	NA	88.86	91.95	90.03

Conclusion

For its exponential expansion, IoT necessitates a network infrastructure that is dependable, dynamic, flexible, quicker, and secure. We presented a unique DL-driven SDN-enabled IoT detection framework in this research to combat sophisticated multivector botnet attacks (i.e., Bot, DDoS, and port scan). The LSTM model was used to propose, implement, and train a network intrusion detection model in this paper. The CICIDS2017 dataset was used to train and test this model. On the CICIDS2017 Dataset, the LSTM model achieved exceptional accuracy with 99.32 for multiclass classification when compared to the latest deep learning methods. Furthermore, the proposed SDN-enabled IoT architecture does not exhaust the underlying IoT resource limited devices. The suggested model beats current benchmarks in terms of detection accuracy after a thorough review and comparison.

With the goal of intelligently securing industrial IoT data, our future research will focus on deep learning fusion and online learning for the network intrusion detection problem.

Declarations

Compliance with Ethical Standards:

Conflict of Interests: The authors have no conflicts of interest to declare that are relevant to the content of this article.

Ethics Approval: This article does not contain any studies with human participants or animals performed by any of the authors.

Data Availability: Data sharing not applicable to this article as no datasets were generated or analysed during the current study.

Acknowledgement: This research is done with the financial support by the Deanship of Scientific Research at King Khalid University under research grant number (RGP.2/241/43).

References

1. Mrabet, H.; Belguith, S.; Alhomoud, A.; Jemai, A. A Survey of IoT Security Based on a Layered Architecture of Sensing and Data Analysis. Sensors 2020, 20, 3625
2. T. Giannetsos, S. Gisdakis, and P. Papadimitratos, “Trustworthy peoplecentric sensing: Privacy, security and user incentives road-map,” in 13th Annual Mediterranean Ad Hoc Networking Workshop, 2014, pp. 39–46.
3. J. Whitefield, L. Chen, T. Giannetsos, S. Schneider, and H. Treharne, “Privacy-enhanced capabilities for vanets using direct anonymous attestation,” in 2017 IEEE Vehicular Networking Conference (VNC), 2017, pp. 123–130.
4. Bhunia, S.S.; Gurusamy, M. Dynamic attack detection and mitigation in IoT using SDN. In Proceedings of the 27th International Telecommunication Networks and Applications Conference (ITNAC), Melbourne, VIC, Australia, 22–24 November 2017; pp. 1–6.
5. Ferdowsi, A.; Saad, W. Deep Learning for Signal Authentication and Security in Massive Internet-of-Things Systems. IEEE Trans. Commun. 2019, 67, 1371–1387.
6. Haller, S.; Karnouskos, S.; Schroth, C. The internet of things in an enterprise context. In Future Internet Symposium; Springer:Berlin/Heidelberg, Germany, 2008; pp. 14–28.
7. Ding, D.; Qing-Long, H.; Yang, X.; Xiaohua, G.; Xian-Ming, Z. A survey on security control and attack detection for industrial cyber-physical systems. Neurocomputing 2018, 275, 1674–1683.
8. Tayfour OE, Marsono MN (2020) Collaborative detection and mitigation of distributed denial-of service attacks on software-defined network. ACM/Springer Mob Netw Appl SI Green Comput Commun 25:1338–1347
9. S. S. Silva, R. M. Silva, R. C. Pinto, and R. M. Salles, “Botnets: A survey,” Computer Networks, vol. 57, no. 2, pp. 378–403, 2013.

10. R. A. Rodr´ıguez-G´omez, G. Maci´a-Fern´andez, and P. Garc´ıa-Teodoro, “Survey and taxonomy of botnet research through life-cycle,” ACM Computing Surveys (CSUR), vol. 45, no. 4, pp. 1–33, 2013.
11. Wu, K.; Chen, Z.; Li, W. A Novel Intrusion Detection Model for a Massive Network Using Convolutional Neural Networks. IEEE Access 2018, 6, 50850–50859
12. A. Akhunzada, A. Gani, N. B. Anuar, A. Abdelaziz, M. K. Khan, A. Hayat, and S. U. Khan, Secure and dependable software de_ned networks," J. Netw. Comput. Appl., vol. 61, pp. 199_221, 2016.
13. Tayfour OE ,Marsono MN . Collaborative detection and mitigation of DDoS in software-defined networks. J. Supercomput. 2021:1–25
14. M. Abadi, U. Erlingsson, I. Goodfellow, H. B. Mcmahan, I. Mironov, N. Papernot, K. Talwar, and L. Zhang, On the protection of pri- vate information in machine learning systems: Two recent approches," in *Proc. IEEE 30th Comput. Secur. Found. Symp. (CSF)*, Aug. 2017, pp. 1_6.
15. M. Rahman Minar and J. Naher, Recent advances in deep learning: An overview," 2018, *arXiv:1807.08169*. [Online]. Available: http://arxiv. org/abs/1807.08169
16. A. Sahl and S. Hasan, "Radiology reports automated annotation performance: rule-based machine learning vs deep learning," 3rd Smart Cities Symposium (SCS 2020), 2020, pp. 433-436, doi: 10.1049/icp.2021.0893.
17. S. Garg, K. Kaur, N. Kumar, and J. J. P. C. Rodrigues, “Hybrid deep-learning-based anomaly detection scheme for suspicious flow detection in SDN: A social multimedia perspective,” IEEE Trans. Multimedia, vol. 21,no. 3, pp. 566–578, Mar. 2019.
18. 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. Springer, 2019, pp. 175–195.
19. S. Maeda, A. Kanai, S. Tanimoto, T. Hatashima, and K. Ohkubo, “A botnet detection method on SDN using deep learning,” in Proc. IEEE Int. Conf. Consum. Electron. (ICCE), Jan. 2019, pp. 1–6.
20. A.R. Narayanadoss, T. Truong-Huu, P.M. Mohan, M. Gurusamy, Crossfire attack detection using deep learning in software defined its networks, in: 2019 IEEE 89th Vehicular Technology Conference (VTC2019-Spring), IEEE, 2019, pp. 1–6
21. C. Li, Y. Wu, X. Yuan, Z. Sun, W. Wang, X. Li, and L. Gong, “Detection and defense of DDoS attack-based on deep learning in OpenFlow-based SDN,” Int. J. Commun. Syst., vol. 31, no. 5, p. e3497, Mar. 2018.
22. T. A. Tang, L. Mhamdi, D. McLernon, S. A. R. Zaidi, and M. Ghogho,“Deep recurrent neural network for intrusion detection in SDN-based networks,” in Proc. 4th IEEE Conf. Netw. Softwarization Workshops (NetSoft), Jun. 2018, pp. 202–206.
23. 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. Electr. Eng. Inf. Commun. Technol. (iCEE-iCT), Sep. 2018, pp. 630–635.
24. Vinayakumar, R.; Soman, K.P.; Poornachandran, P. Evaluation of Recurrent Neural Network and its Variants for Intrusion Detection System (IDS). Int. J. Inf. Syst. Model. Des. 2017, 8, 43–63.

25. Y. Li, Y. Xu, Z. Liu, H. Hou, Y. Zheng, Y. Xin, Y. Zhao, L. Cui, Robust detection for network intrusion of industrial IoT based on multi-CNN fusion, Measurement 154 (2020) 107450
26. Liaqat, S.; Akhunzada, A.; Shaikh, F.S.; Giannetsos, A.; Jan, M.A. SDN orchestration to combat evolving cyber threats in Internet of Medical Things (IoMT). Comput. Commun. 2020, 160, 697–705.
27. Abdullah Soliman Alshra'a, Ahmad Farhat, Jochen Seitz,Deep Learning Algorithms for Detecting Denial of Service Attacks in Software-Defined Networks, Procedia Computer Science, Volume 191, 2021, Pages 254-263, ISSN 1877-0509, https://doi.org/10.1016/j.procs.2021.07.032
28. I. Sharafaldin, A. Habibi Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in Proc. 4th Int. Conf. Inf. Syst. Secur. Privacy, 2018,pp. 108–116.
29. S. Lang, F. Bravo-Marquez, C. Beckham, M. Hall, and E.Frank WekaDeeplearning4j: a Deep Learning Package for Weka based on DeepLearning4j, In *Knowledge-Based Systems*, Volume 178, 15 August 2019, Pages 48-50. DOI: 10.1016/j.knosys.2019.04.013
30. M. Tan, A. Iacovazzi, N.-M. M. Cheung, and Y. Elovici, "A neural attention model for real-time network intrusion detection," in 2019 IEEE 44th Conference on Local Computer Networks (LCN). IEEE, 2019, pp.291–299.

Figures

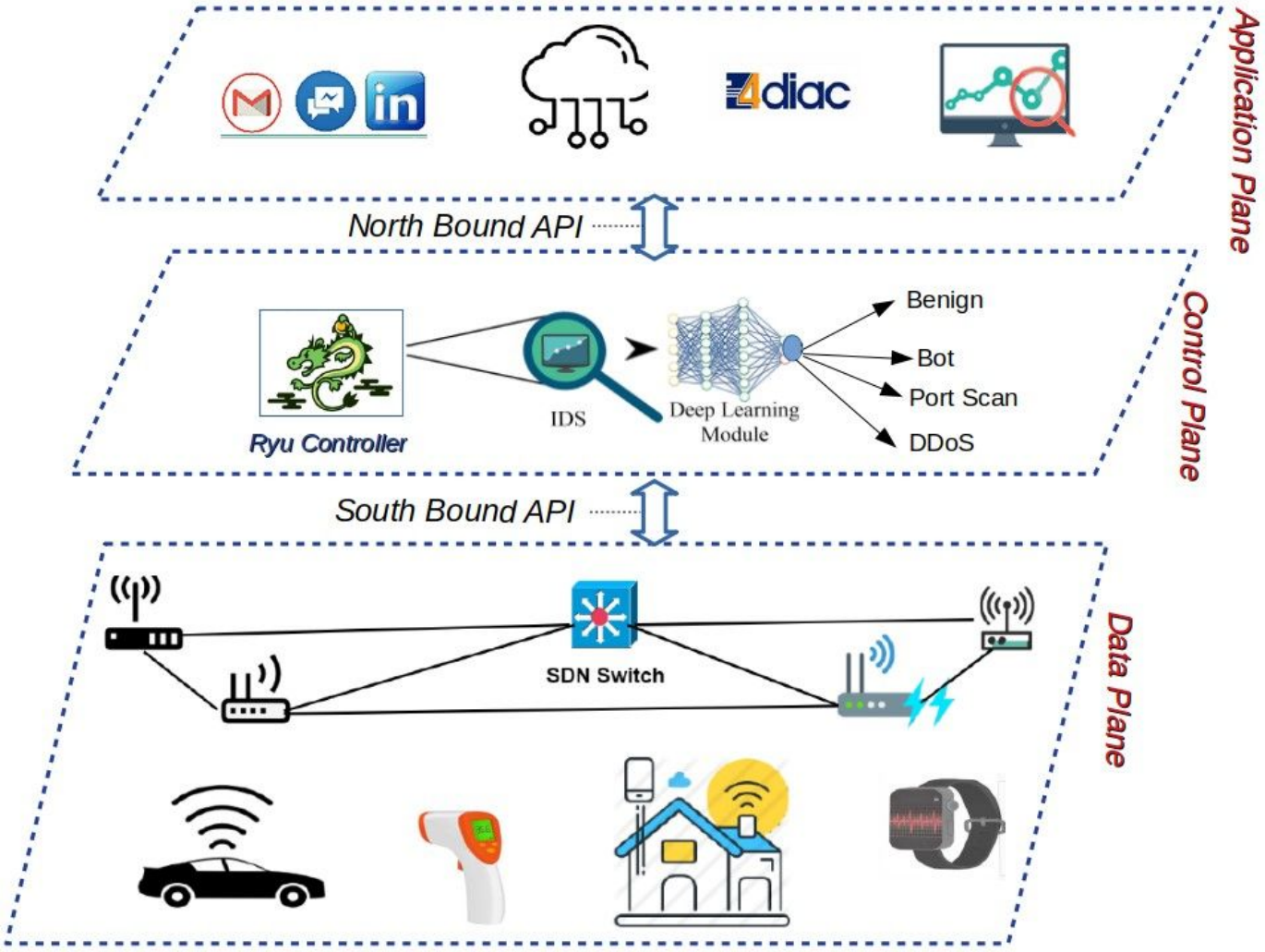

Figure 1

SD-IoT proposed architecture

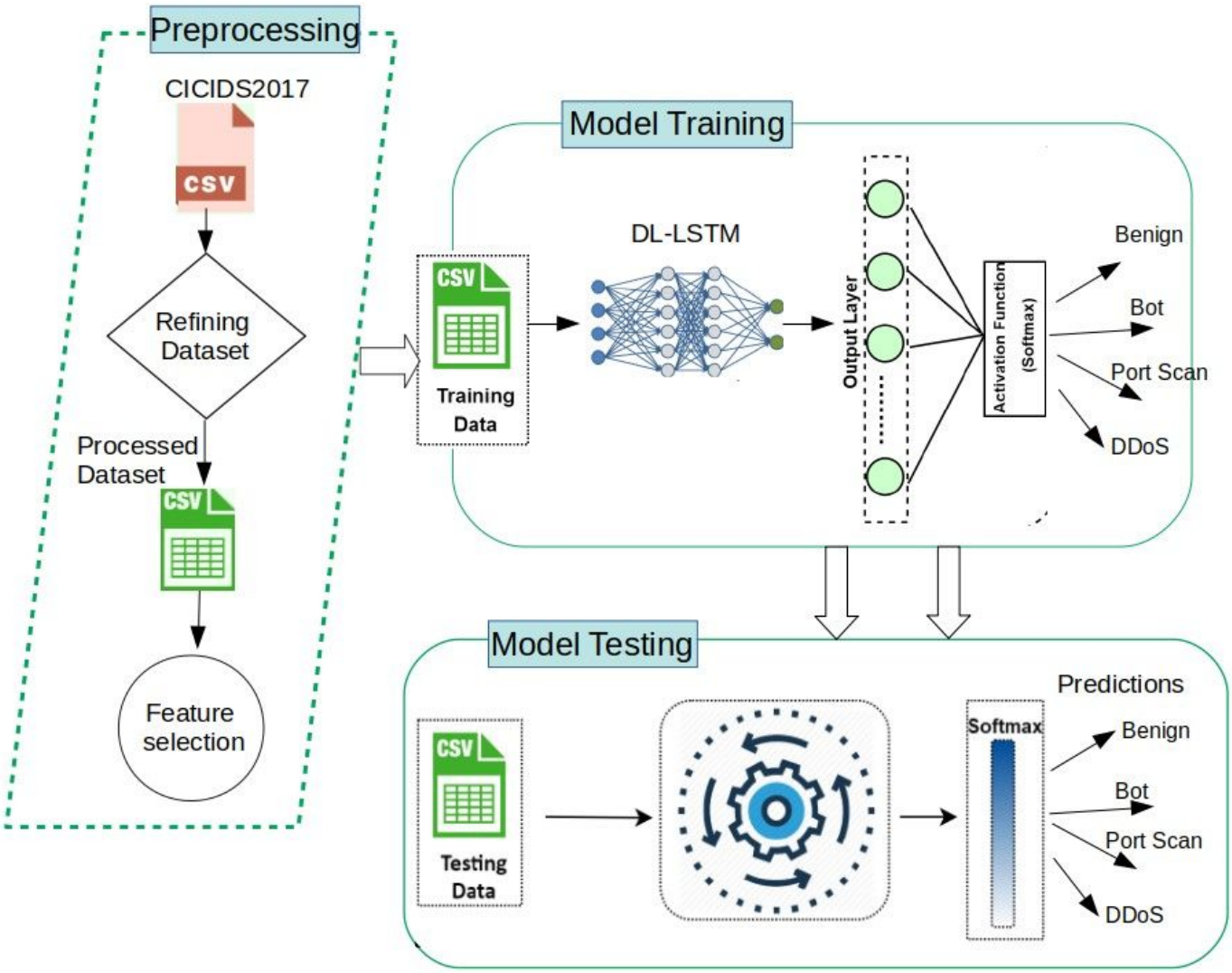

Figure 2

LSTM Framework

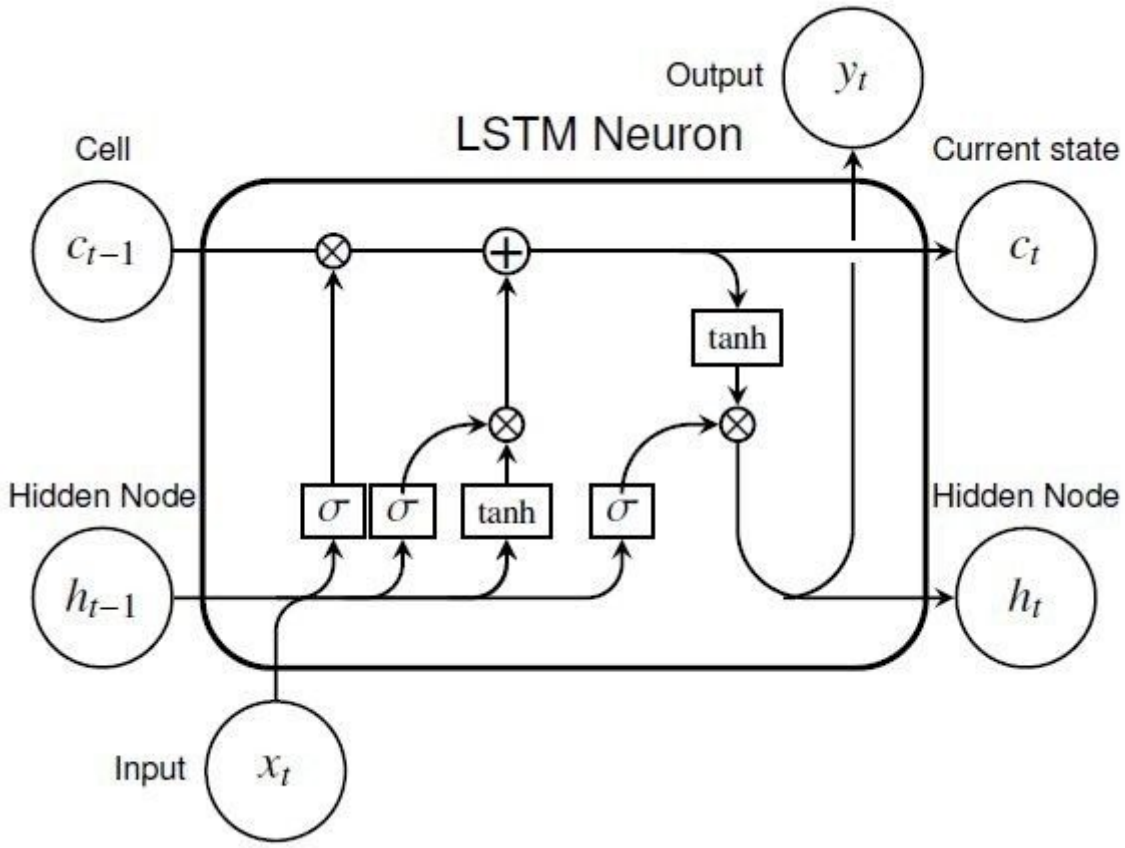

Figure 3

LSTM Neuron [27]

Sr. No	Features Names	Sr. No	Features Names	Sr. No	Features
1	Source IP	29	Fwd IAT Max	57	Average Packet Size
2	Source Port	30	Fwd IAT Min	58	Avg Fwd Segment Size
3	Destination IP	31	Bwd IAT Total	59	Avg Bwd Segment Size
4	Destination Port	32	Bwd IAT Mean	60	Fwd Header Length
5	Protocol	33	Bwd IAT Std	61	Fwd Avg Bytes/Bulk
6	Timestamp	34	Bwd IAT Max	62	Fwd Avg Packets/Bulk
7	Flow Duration	35	Bwd IAT Min	63	Fwd Avg Bulk Rate
8	Total Fwd Packets	36	Fwd PSH Flags	64	Bwd Avg Bytes/Bulk
9	Total Backward Packets	37	Bwd PSH Flags	65	Bwd Avg Packets/Bulk
10	Total Length of Fwd Packets	38	Fwd URG Flags	66	Bwd Avg Bulk Rate
11	Total Length of Bwd Packets	39	Bwd URG Flags	67	Subflow Fwd Packets
12	Fwd Packet Length Max	40	Bwd Header Length	68	Subflow Fwd Bytes
13	Fwd Packet Length Min	41	Fwd Packets/s	69	Subflow Bwd Packets
14	Fwd Packet Length Mean	42	Bwd Packets/s	70	Subflow Bwd Bytes
15	Fwd Packet Length Std	43	Min Packet Length	71	Init_Win_bytes_forward
16	Bwd Packet Length Max	44	Max Packet Length	72	Init_Win_bytes_backward
17	Bwd Packet Length Min	45	Packet Length Mean	73	act_data_pkt_fwd
18	Bwd Packet Length Mean	46	Packet Length Std	74	min_seg_size_forward
19	Bwd Packet Length Std	47	Packet Length Variance	75	Active Mean
20	Flow Bytes/s	48	FIN Flag Count	76	Active Std
21	Flow Packets/s	49	SYN Flag Count	77	Active Max
22	Flow IAT Mean	50	RST Flag Count	78	Active Min
23	Flow IAT Std	51	PSH Flag Count	79	Idle Mean
24	Flow IAT Max	52	ACK Flag Count	80	Idle Std
25	Flow IAT Min	53	URG Flag Count	81	Idle Max
26	Fwd IAT Total	54	CWE Flag Count	82	Idle Min
27	Fwd IAT Mean	55	ECE Flag Count	83	Label
28	Fwd IAT Std	56	Down/Up Ratio		

Figure 4

CICIDS2017 Dataset-Features

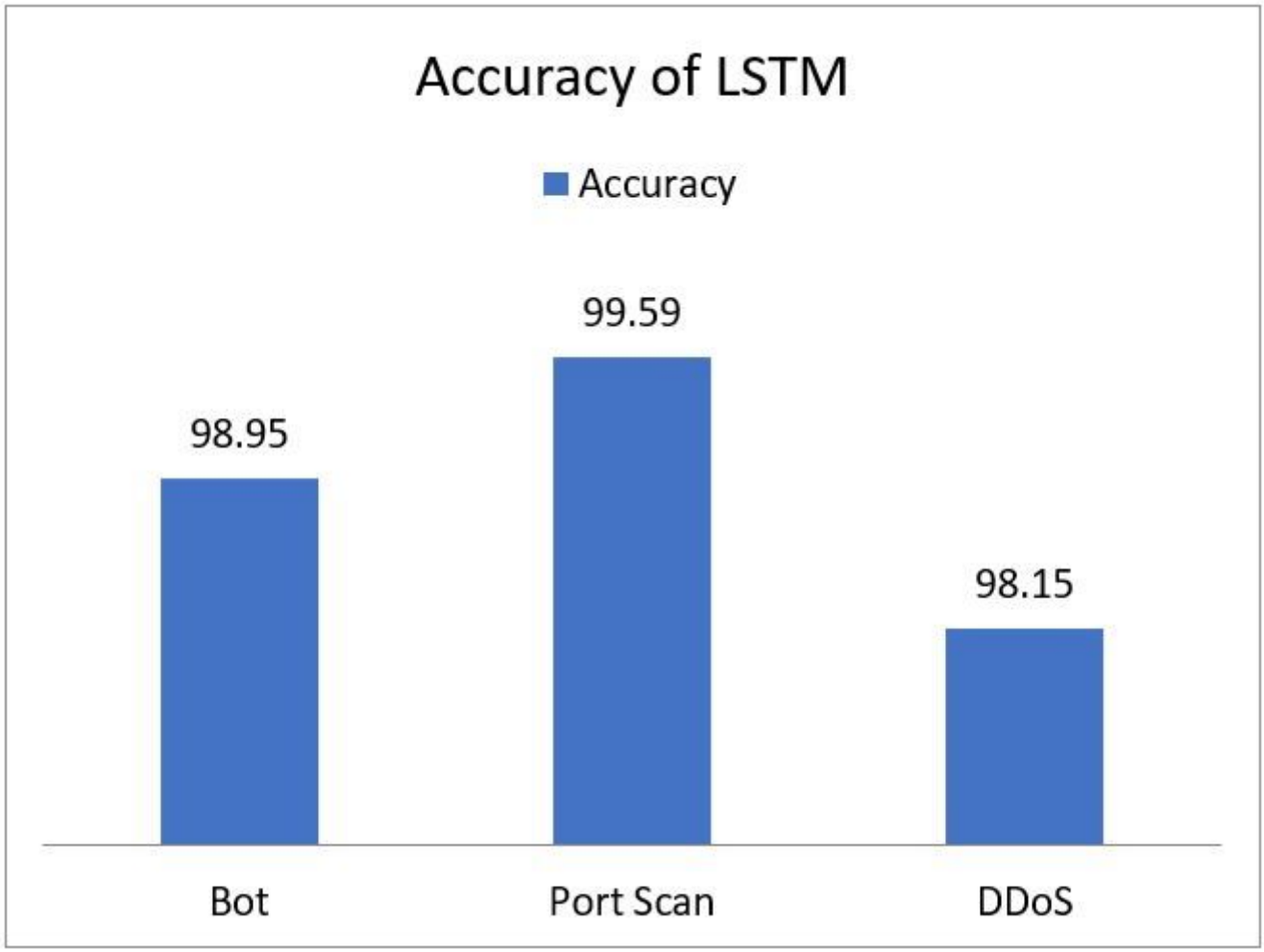

Figure 5

per-class accuracy

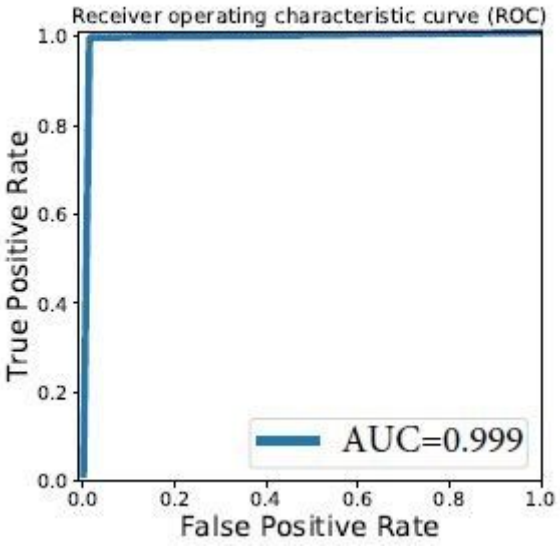

Figure 6

ROC curve for LSTM Model