South Asian Journal of Science and Technology

Open Access Full Text Article 152(2025) 123-128 DOI: 10.26524/sajet.2025.15.13

Using Deep Learning an Effiecient Bot Attack Detection Methods

Dr.A.Chandrasekar[1] , K.Yamuna[2]

[1]Professor, [2]PG Scholar-First Year, Department of Computer Science and Engineering, Nandha College of Technology, Perundurai , Tamilnadu , India.

Corresponding author.

Correspondence: K.Yamuna

Article info

Received 24th December 2024 Received in revised form 4 April 2025

Accepted 6 May 2025

https://sajet.in/index.php/journal/article/view/328

Abstract

Deep Learning (DL) is an effective way to detect botnet attacks. However, the amount of network traffic data and the required memory space are usually large. Therefore, it is almost impossible to use the DL method on memory-restricted IoT devices. In this paper, we reduce the size of the IoT network traffic data feature using the Long Short-Term Short-Term Memory Autoencoder (LAE) codec section. In order to classify network traffic samples correctly, we analyze long-term variables related to low-dimensional feature produced by LAE using Bi-directional Long Short-Term Memory (BLSTM). Comprehensive testing was performed with BoT-IoT databases to confirm the effectiveness of the proposed DL hybrid method. The results show that LAE significantly reduced the memory space required for data storage of large network traffic by 91.89%, and exceeded the standard features of reducing feature by 18.92 -27.03%. Despite the significant reduction in feature size, the deep BLSTM model shows strength against low model equity and over-equilibrium. It also acquires a good ability to adapt to the conditions of binary classification.

1. INTRODUCTION

With the selection of the active feature and the accurate detection of Bot-IoT attacks in the IoT network area using a new development database. The database includes Internet of Things, with normal traffic flow and several online attacks on botnets attacks. In order to track accurate traffic and improve an active database, a virtual test bed is used for the development of this database with functional information features. Similarly, in order to improve the performance of the machine learning model and the effective guessing model, many features were extracted and added to a set of extracted features. However, for best performance results, the extracted features are labeled, such as attack flow, stages, and subdivisions. Today, Internet of Things (IoT) technology is growing exponentially day by day, and every minute, more and more devices are connected to this technology. By using this technology, daily life becomes easier and more organized. For example, initially, IoT technology was limited to small offices and apartments, but today, IoT technology is integrated into industries to make it more reliable and time-saving. However, IoT technology is becoming an integral part of our daily lives. By 2021, IoT technology will grow, and more than 27 million IoT devices will be connected, which will be a major change in the world of IoT technology. With the rapid development and popularity of Internet of Things (IoT) devices, an increasing number of Internet attacks have targeted these devices. It is said that most IoT site attacks are botnet-based attacks. Many security vulnerabilities still exist on IoT devices because most of them do not have enough memory and a calculator for robust security systems. In addition, many existing law-based detection systems can be blocked by attackers. In this study, we proposed a botnet detection (ML) framework based on botnet-based detection systems. An effective feature selection approach is adopted for the efficient

use of the weight loss system. Botnet is a network of multiple bots designed to perform malicious activity on a targeted network controlled using a single control unit and control unit called a botmaster. Infected computer bots are remotely controlled by the botmaster without the sign of a hack and are used to perform malicious activities.

Objectives:

- The main purpose of our project is to detect network attacks
- Implementing a factor size reduction such as a Partial Component Analysis (PCA), to reduce the amount of size in the database.
- Using the default encoder for compressing raw data.
- Using a deep learning algorithm like LSTM and CNN, for better performance.
- Improving performance analysis.

Problem Statement:

The most common way to detect botnets is to track and analyze the attack itself while the standard security solutions provide visibility and determine which attacks are coming from the botnet. Botnets can contain thousands of hackers and can be used to attack a variety of Internet-based attacks, especially flood-targeted networks and devices with too much traffic and data theft by hackers.

2 System Proposal

2.1 Existing System

Deep Learning (DL) is an effective way to detect botnet attacks. However, the amount of network traffic data and the required memory space are usually large. Therefore, it is almost impossible to use the DL method on memory-restricted IoT devices. In this paper, we reduce the size of the IoT network traffic data feature. The system is designed to minimize feature size. Then, use algorithms for advanced reading algorithms such as LSTM and default encoder. Also used for Bidirectional LSTM. Comprehensive testing was performed with BoT-IoT databases to confirm the effectiveness of the proposed DL hybrid method. The results show that LAE significantly reduced the memory space required for data storage of large network traffic by 91.89%, and exceeded modern means of reducing modern features by 18.92–27.03%. Despite the significant reduction in feature size, the deep BLSTM model shows strength against the model of inequality and over-equity. It also acquires a good ability to adapt to the conditions of binary separation with multiple categories.

2.2 Proposed System

In this program, a Bot-Iot data set was taken as installed. Input data is taken from the database. Then, we must take the initiative to process the data. in this step, we should manage the missing values to avoid incorrect guessing, encoding the input data label and to normalize / measure input data. Then, we should apply a reduction in feature size such as Key Segment Analysis (PCA) is one of the most common line conversion methods while kernel methods, spectral methods and DL methods use non-line conversion techniques. Next, we should use the default embedded method of the unlocked DL method that generates a hidden representation of the input data in the hidden layer. Different configurations for the default encoder are suggested to reduce the feature size in the most popular network login databases. Then, we have to use in-depth learning algorithms like Long-Term Memory (LSTM) and Convolutional Neural Network (CNN). Finally, test results show that performance metrics such as accuracy, precision, memory and confusion matrix.

3 Implementation

3.1 Data Selection:

Input data collected from a database. In our process, a Bot-IoT database is used. Data sorting is the process of getting malicious traffic. The database includes Internet of Things, with normal traffic flow and several online attacks on botnets attacks. In order to track accurate traffic and improve an active

database, a virtual test bed is used for the development of this database with functional information features. Similarly, in order to improve the performance of the in-depth learning model and the effective guessing model, many features were extracted and added to a set of extracted features. However, for best performance results, the extracted features are labelled, such as attack flow, stages, and subdivisions.

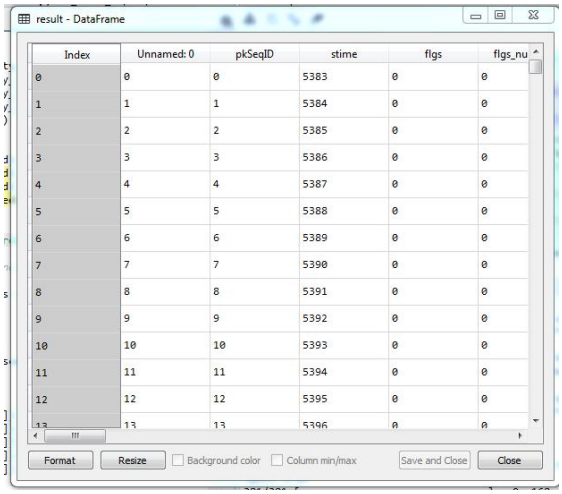

FIGURE 4.2.1: DATA SELECTION

3.2 Data Pre-processing:

Pre-data processing is the process of extracting unwanted data from the database. Pre-processing data conversion functions are used to convert databases into machine-readable formats. This step includes cleaning up the database by removing unimportant or damaged data that may affect the accuracy of the database, making it more efficient. Missing data deletion Phase data Missing data: In this process, empty values such as missing values and Nan values are replaced by 0.Missing and duplicate values were extracted and data deleted from any variables.

Phase data encoding: That category data is defined as a variable with a limited set of label values.That most machine learning algorithms require numerical input and outputvariables.

```
dtype: int64

-------------------------Before Label Encoding--------------------
   Unnamed: 0  pkSeqID         stime  ... attack  category subcategory
0     1650261  1650261  1.528103e+09  ...      1      DDoS        HTTP
1     1650262  1650262  1.528103e+09  ...      1      DDoS        HTTP
2     1650263  1650263  1.528103e+09  ...      1      DDoS        HTTP
3     1650264  1650264  1.528103e+09  ...      1      DDoS        HTTP
4     1650265  1650265  1.528103e+09  ...      1      DDoS        HTTP
5     1650266  1650266  1.528103e+09  ...      1      DDoS        HTTP
6     1650267  1650267  1.528103e+09  ...      1      DDoS        HTTP
7     1650268  1650268  1.528103e+09  ...      1      DDoS        HTTP
8     1650269  1650269  1.528103e+09  ...      1      DDoS        HTTP
9     1650270  1650270  1.528103e+09  ...      1      DDoS        HTTP

[10 rows x 47 columns]
```

FIGURE 4.2.2.1: BEFORE LABEL ENCODING

```
------------------------After Label Encoding----------------------

   Unnamed: 0  pkSeqID  stime  ...  attack  category  subcategory
0           0        0   5383  ...       1         0            0
1           1        1   5384  ...       1         0            0
2           2        2   5385  ...       1         0            0
3           3        3   5386  ...       1         0            0
4           4        4   5387  ...       1         0            0
5           5        5   5388  ...       1         0            0
6           6        6   5389  ...       1         0            0
7           7        7   5390  ...       1         0            0
8           8        8   5391  ...       1         0            0
9           9        9   5392  ...       1         0            0
```

FIGURE 4.2.2.2: AFTER LABEL ENCODING

3.3 Data Normalization:

Python provides a pre-processing library, which contains the custom function to make the data normal. It takes the same members as input and makes its values normal between 00 and 11. It then returns the output list with the same size as input.

Frequency measurement for each input varies to a range of 0-1, which is a floating point value range when we have high accuracy. The measurement is different for each input by subtracting meaning (centering) and dividing by standard deviation so that the distribution has zero meaning and standard deviation for each.

3.4 Dimensionality Reduction

The number of input features, variables, or columns present in a given dataset, this is known as dimensionality, and the process of reducing these features is known as dimensionality reduction. Dimensional reduction techniques can be defined as a method of converting datasets of higher dimensions to datasets of lower dimensions ensuring that it provides uniform information. In this step, we have to use PCA (Principle Component Analysis). PCA is used for visualization, noise filtering and feature extraction. PCA is a dimensionality reduction that identifies significant relationships in our data, transforming existing data based on these relationships. Then scales the importance of these relationships so that we can keep the most important relationships and discard the others. These techniques are widely used in machine learning to obtain a better fit predictive model while solving classification and regression problems.

4. Result Generation

The final result will be generated on the basis of overall classification and prediction. The performance of this proposed approach is evaluated using certain measures, such as,

The accuracy of the classifier refers to the capability of the classifier. It correctly predicts class labels, and the accuracy of the predictor refers to how well a given predictor can predict the value of the predicted attribute for new data.

AC= (TP+TN)/ (TP+TN+FP+FN)

Precision is defined as the number of true positives divided by the number of true positives and the number of false positives.

Precision=TP/ (TP+FP)

nnnThe recall is the number of correct results divided by the number of results that should have been returned. In binary classification, recall is called sensitivity.

Recall=TP/ (TP+FN)

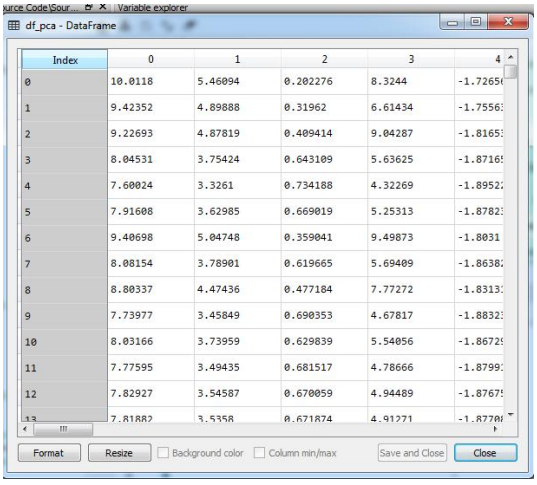

FIGURE 4.2.7.1: DF-PCA DATAFRAME

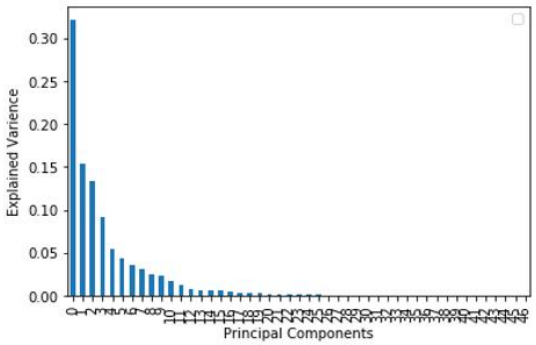

FIGURE 4.2.7.2: Explained Variance vs Principal Components

5. Conclusion& Future Enhancement

This system was proposed for efficient botnet detection in IoT networks using deep learning algorithms such as LSTM and CNN. The effectiveness of this method was validated by conducting extensive experiments with the most relevant publicly available datasets (BOT-IoT) in binary and multi-class classification scenarios. In this context, Bot IoT was used as a dataset because of its regular updates, wide attack diversity, and different network protocols. We evaluate our proposed approach using the Bot-IoT dataset. The analysis of experimental results showed that our proposed method is efficient and can achieve better performance results on average than LSTM. As a future work, it will be interesting to evaluate the performance of some unsupervised algorithms. In addition, we implemented various deep and machine learning algorithms independently of each other. In the future, we should combine various machine learning and deep learning

Reference

1. J. Qiu, Z. Tian, C. Du, Q. Zuo, S. Su, and B. Fang, "A survey on access control in the age of internet of things," IEEE Internet of Things Journal, 2020.
2. Y. N. Soe, Y. Feng, P. I. Santosa, R. Hartanto, and K. Sakurai, "Implementing lightweight iot-ids on raspberry pi using correlation based feature selection and its performance evaluation," in International Conference on Advanced Information Networking and Applications. Springer, 2019, pp. 458–469.
3. K. Lab. (2019) Amount of malware targeting smart devices more than doubled in. [Online].
4. J. Qiu, L. Du, D. Zhang, S. Su, and Z. Tian, "Nei-tte: Intelligent traffic time estimation based on fine-grained time derivation of road segments for smart city," IEEE Transactions on Industrial Informatics, 2019.
5. J. P. Anderson, "Computer security threat monitoring and surveillance, 1980. lastaccessed: Novmeber 30, 2008."
6. D. E. Denning, "An intrusion-detection model," IEEE Transactions on software engineering, no. 2, pp. 222–232, 1987.
7. L. Wu, X. Du, W. Wang, and B. Lin, "An out-of-band authentication scheme for internet of things using blockchain technology," in 2018 International Conference on Computing, Networking and Communications (ICNC). IEEE, 2018, pp. 769–773.
8. Z. Tian, X. Gao, S. Su, and J. Qiu, "Vcash: A novel reputation framework for identifying denial of traffic service in internet of connected vehicles," IEEE Internet of Things Journal, vol. 7, no. 5, pp. 3901– 3909, May 2020.
9. S. Alharbi, P. Rodriguez, R. Maharaja, P. Iyer, N. Bose, and Z. Ye, "Focus: A fog computing-based security system for the internet of things," in 2018 15th IEEE Annual Consumer Communications & Networking Conference (CCNC). IEEE, 2018, pp. 1–5.
10. Z. Tian, C. Luo, J. Qiu, X. Du, and M. Guizani, "A distributed deep learning system for web attack detection on edge devices," IEEE Transactions on Industrial Informatics, 2020. Vol 16(3): 1963-1971.

11. D. Ventura, D. Casado-Mansilla, J. López-de Armentia, P. Garaizar, D. López-de Ipina, and V. Catania, "Ariima: a real iot implementation of a machine-learning architecture for reducing energy consumption," in International Conference on Ubiquitous Computing and Ambient Intelligence. Springer, 2014, pp. 444–451.
12. R. Xue, L. Wang, and J. Chen, "Using the iot to construct ubiquitous learning environment," in 2011 Second International Conference on Mechanic Automation and Control Engineering. IEEE, 2011, pp. 7878– 7880.
13. M. A. Alsheikh, S. Lin, D. Niyato, and H.-P. Tan, "Machine learning in wireless sensor networks: Algorithms, strategies, and applications," IEEE Communications Surveys & Tutorials, vol. 16, no. 4, pp. 1996– 2018, 2014.
14. M. Shafiq, X. Yu, A. A. Laghari, and D. Wang, "Effective feature selection for 5g im applications traffic classification," Mobile Information Systems, vol. 2017, 2017.
15. M. Shafiq, X. Yu, A. K. Bashir, H. N. Chaudhry, and D. Wang, "A machine learning approach for feature selection traffic classification using security analysis," The Journal of Supercomputing, vol. 74, no. 10, pp. 4867–4892, 2018.
16. K. Anoh, A. Ikpehai, D. Bajovic, O. Jogunola, B. Adebisi, D. Vukobratovic, and M. Hammoudeh, "Virtual microgrids: a management concept for peer-to-peer energy trading," in Proceedings of the 2nd International Conference on Future Networks and Distributed Systems, 2018, pp. 1–5.
17. O. Akmandor, Y. Hongxu, and N. K. Jha, "Smart, secure, yet energyefficient, internet-of-things sensors," IEEE Transactions on Multi-Scale Computing Systems, vol. 4, no. 4, pp. 914–930, 2018.
18. Z. Wang, Y. Liu, Z. Ma, X. Liu, and J. Ma, "Lipsg: Lightweight privacypreserving q-learning based energy management for the iot-enable smart grid," IEEE Internet of Things Journal, vol. 7, no. 5, pp. 3935–3947, 2020.
19. M. R. Palattella, M. Dohler, A. Grieco, G. Rizzo, J. Torsner, T. Engel, and L. Ladid, "Internet of things in the 5g era: Enablers, architecture, and business models," IEEE Journal on Selected Areas in Communications, vol. 34, no. 3, pp. 510–527, 2016.
20. V.S. Suresh kumar "Frequent Pattern Complex query management using FIUT Approach", South Asian Journal of Engineering and Technology, pp: 300-304, issue 204, volume 202, 2018
21. Sureshkumar V S, Chandrasekar A," Fuzzy-GA Optimized Multi-Cloud Multi-Task Scheduler For Cloud Storage And Service Applications" International Journal of Scientific & Engineering Research, Vol.04, Issue.3,pp-1-7, 2013
22. E.Prabhakar,V.S.Sureshkumar, Dr.S.Nandagopal, C.R.Dhivyaa, Mining Better Advertisement Tool for Government Schemes Using Machine learning " , International Journal of Psychosocial Rehabilitation, Vol.23,Issue.4, pp. 1122-1135, 2019
23. Suresh kumar V S ,Thiruvankatasamy S, Sudhakar R, "Optimized Multicloud Multitask Scheduler For Cloud Storage And Service By Genetic Algorithm And Rank Selection Method", Vol.3,Issue.2, pp.1-6, 2014
24. Nandagopal S, Malathi T, "Enhanced Slicing Technique for Improving Accuracy in Crowd Sourcing Database", International Journal of Innovative Research in Science, Engineering and Technology, Vol.3,Issue.1, pp.278-284, 2014
25. V.S. Suresh kumar "E-Farming by means of E-Mandi Process", International Journal of Research and Advanced Development (IJRAD), ISSN: 2581-4451, pp: 55-57, Issue 6, volume 2, 2019
26. Dr.C.R. Dhivyaa, R. Sudhakar, K. Nithya and E. Prabhakar "Performance Analysis of Convolutional NeuralNetwork for Retinal Image Classification", International Journal of Psychosocial Rehabilitation, Vol. 23, no.4, pp.1149-1159,November 2019.
27. V.S. Sureshkumar K. Boobesh , G. Dhatchayan , S. Karthick raja , R. Ajayeswaran" A High-Efficient Joint 'Cloud-Edge' Aware Strategy for Task Deployment and Load Balancing" south asian journal of engineering and technology, Vol. 13, issue no.1, pp. 22-38,November 2024.
28. P.Dhatchana Moorthy, J.Jenshya, V.S Suresh kumar, F.Christopher Jerome, A.S Mahant, N. M Vallarasu. "Early Breast Cancer Detection and Diagnosis Using An Efficient net-Based Predictive System", Second International Conference on Advances in Modern Age Technologies for Health and Engineering Science, 2025.