Electronic Engineering and Informatics
G. Izat Rashed (Ed.)

doi:10.3233/ATDE240122

Improved IoT Network Malicious Traffic Detection Method Based on Extreme Value Theory

Zezhong ZHANG[1], Jianxin ZHOU[2] and Ning ZHOU[3]
School of Information Engineering, Wuhan University of Technology, Wuhan, 430070, China

Abstract. Correctly identifying malicious traffic has always been one of the most critical issues in cybersecurity. In recent years, with the increasing number of Internet of Things (IoT) devices, attacks on IoT networks are also increasing gradually, and the methods of attack are constantly diversifying. Existing research on distinguishing malicious traffic is limited due to a lack of previous datasets focused on conventional networks instead of IoT networks. Additionally, the recognition ability of models is determined by the categories used in the training set, making it unable to identify samples outside the training set categories correctly. In this paper, we propose a deep learning method based on extreme value theory (EVT) by improving existed OpenMax method with focal loss function and other techniques. This method can reject traffic data not present in the training set and classify it as unknown. Experimental results on the IoT-23 dataset demonstrate that, among all relevant papers that we are aware of using this dataset, the proposed method stands out as the first in maintaining multi-classification accuracy for known traffic while rejecting over 90% of unknown traffic.

Keywords. malicious traffic dectection; traffic classification; deep learning; extreme value theory; open set recognition

1 Introduction

Malicious traffic detection is crucial for identifying and distinguishing between benign and malicious network traffic. It enables internet service providers to safeguard their networks and users. The research in this field has a rich history spanning over two decades. The DARPA intrusion detection evaluation dataset, released by The Cyber Systems and Technology Group in 1998 [1], serves as the initial benchmark dataset. The results indicated the need for research focused on developing new attack detection techniques rather than extending existing rule-based approaches. In the past decade, Wang pioneered deep learning algorithms to classify network traffic [2]. Their framework employed Artificial Neural Network and Stacked Auto Encoder to classify private datasets. Inspired by this, Wang et al. deployed a 2-dimensional convolutional

[1] Zezhong ZHANG, School of Information Engineering, Wuhan University of Technology, e-mail: 276218@whut.edu.cn
[2] Corresponding author: Jianxin ZHOU, School of Information Engineering, Wuhan University of Technology, e-mail: zjx@whut.edu.cn
[3] Ning ZHOU, School of Information Engineering, Wuhan University of Technology, e-mail: zhouning@whut.edu.cn

neural network to classify traffic from different applications [3].

The proliferation of IoT devices has garnered significant attention. The increasing number of IoT devices, and their inherent vulnerabilities, expose them to cybercriminal targeting. To address these security concerns, researchers have published datasets such as Bot-IoT [4], TON-IoT [5], and IoT-23 [6]. IoT-23, which aims explicitly at IoT devices, includes 20 malware captures on IoT devices and 3 captures representing benign traffic. Consequently, this paper focuses on the IoT-23 dataset and developing a malicious traffic detection model, and achieves an accuracy of 99.7% for multiple classifications when classifying only known classes. While adding two unknown classes of traffic to the test set, the rejection rate of the unknown class traffic exceeds 90% after improving the loss function and introducing the EVT-based post-processing algorithm. The t-SNE analysis shows that the improved loss can effectively increase the inter-class distance and make the class boundaries clearer.

2 Related Works

2.1 OpenMax

Open set recognition (OSR) is a challenging problem in machine learning, where the goal is to accurately classify unknowns that do not belong to any predefined classes. To address this problem, Bendale and Boult proposed the OpenMax layer [7]. OpenMax refers to the output of the penultimate layer in a classifier as the Activation Vector (AV) and assumes that AVs generated by samples belonging to the same class are similar. Based on the Euclidean distance between a sample's AV and the Mean AV (MAV) of its corresponding class, the authors fit a Weibull distribution. They then use the cumulative distribution function (CDF) of the Weibull distribution to determine the probability of a sample belonging to that class. During testing, the AV scores of the top α classes are recalibrated. The difference between the sum of the activation vector scores before and after recalibration is allocated to the unknown class. By utilizing EVT, the OpenMax approach provides a preliminary solution to the OSR problem at a lower cost.

2.2 Focal Loss

Focal Loss [8] is a loss function used in object detection tasks, specifically designed for imbalanced classes. It addresses the problem by assigning higher weights to misclassified examples from the minority class, emphasizing their importance during training. The key concept is the "focusing parameter," which down-weights easy examples and focuses more on hard examples. Focal Loss combines the cross-entropy term with the focusing parameter, providing a balance between easy and hard examples. It has proven effective in improving the performance of object detection models.

3 Methods

Existing models in research are constrained by the training dataset and can only recognize categories within the training set. Our paper leverages the research achievements in the field of open-set recognition in computer vision and improves upon them to address this issue. In addition, we considered the characteristics of IoT network

traffic, so the model proposed in our paper aims to achieve low computational overhead and fast inference speed.

3.1 Dataset preprocessing and feature selection

The IoT-23 dataset contains over 200 million bidirectional connections represented as flows with a 5-tuple. Each flow consists of 20 fields, including timestamp, flow details, packet information, connection status, history and so on. After preprocessing, the input data has a size of 44 for the neural network. We analysed the labels and focused on categories with over 10, 000 samples, selecting the category that covers the most samples for overlapping categories. The IoT-23 category has extreme class imbalance, addressed using random undersampling and SMOTE upsampling techniques. The goal was to maintain a 10:1 imbalance ratio between the largest and smallest categories. Table 1 presents the selected categories and their sample sizes.

Table 1. Selected categories and sample size.

Categories	Definition	Sample size
Benign	No malicious activities were found in connections.	3, 046, 929
Ddos	A ddos attack is executed by the infected device.	3, 046, 929
Part Of Horizontal Port Scan	The connections are used to do a horizontal port scan to gather information to perform further attacks.	3, 046, 929
Okiru	Connections have features of an Okiru botnet.	383, 297
C&C-Heartbeat	Packets are used to keep track of the infected host by the server.	304, 692
Attack	Some type of attack from the infected device to another host.	304, 692

3.2 Model structure

The proposed model structure that can reject unknown classes is illustrated in Figure 1. We input traffic data of length 44 into a two-layer LSTM network. Subsequently, a scaled dot-product self-attention mechanism is applied to the output data of length 120. Following this, the data sequentially passes through three linear layers with output dimensions of 40, 20, and 4, respectively. Rectified Linear Unit (ReLU) is employed between each pair of linear layers. Then, we used EVT to give the model the ability to reject unknown classes. Additionally, we applied Focal loss to train the model, which resulted in larger MAV distances and enhanced discriminability of unknown classes.

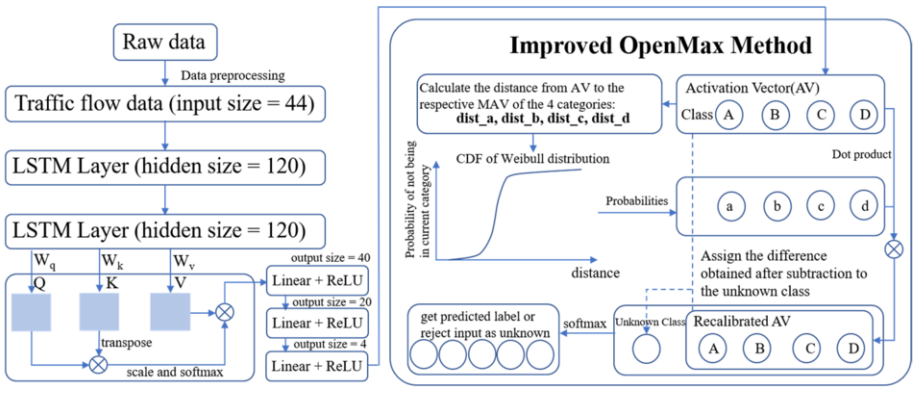

Figure 1. Model structure

3.3 Recognize and reject unknown samples

The existing OpenMax method provides a way to reject unknown inputs. The penultimate layer's output (AV) is used by the OpenMax method. Each class is represented by a point, MAV, which is the mean of correctly classified training examples. The distance between MAV and an input sample is measured to recognize outliers. It considers samples with larger distances to MAV as having a higher risk of not belonging to that class and the CDF of the Weibull distribution to quantify the probability of not belonging to a particular class. Based on this probability, the unknown class is assigned a score in AV, multiplied by the probability of not belonging to that class. The score for the current class is then adjusted as the residual value. This process is repeated for the largest α values in AV, generating a new output containing scores for the unknown class. Finally, a SoftMax function is applied to estimate the probability of the unknown class. However, if one or some of the scores in the activation vector is/are negative and the corresponding correction factor is/are less than 1, then the score for that category is corrected to be higher, while the score assigned to the unknown category becomes lower or even negative. Therefore, we apply Eq. (1) to each value in AV that is negative.

$$v_{new}(x) = e^{v(x)-v_{\min}} + \left|v_{\min}\right| \tag{1}$$

In Eq. (1), v_{min} represents the minimum value in AV for a sample, v(x) represents any value in AV that is less than 0, and $v_{new}(x)$ will be used to replace the v(x) in AV. This way, we obtain an improved OpenMax method, and its detailed procedure can be found in Table 2.

Table 2. Improved OpenMax algorithm.

Algorithm 1. Improved OpenMax probability estimation with the rejection of unknown inputs.

Require: Activation Vector (AV) for $\mathbf{v}(\mathbf{x}) = v_1(x),...,v_N(x)$

Require: Weibull models $\rho_j = (\tau_j, \lambda_j, \kappa_j)$, MAV μ_j and α, the number of "top" classes to revise

1: Let $s(i) = \text{argsort}(v_j(x))$; Let $\omega_j = 1$

2: **for** $i = 1,...,\alpha$ **do**

3: $$\omega_{s(i)}(x) = 1 - \frac{\alpha - i}{\alpha} e^{\rho_{s(i)}(x)}$$

4: **end for**

5: Revise $\mathbf{v}(\mathbf{x})$ by applying (1)

6: Recalibrate AV $\hat{v}(x) = \mathbf{v}_{new}(\mathbf{x}) \circ \omega(x)$

7: Define $\hat{v}_0(x) = \sum_i v_i(x)(1 - \omega_i(x))$

8: Applied SoftMax $\hat{P}(y = j \left| \mathbf{x}) = \frac{e^{\hat{\mathbf{v}}_j(\mathbf{x})}}{\sum_{i=0}^{N} e^{\hat{\mathbf{v}}_i(\mathbf{x})}}$

9: Let $y^* = \text{argmax}_j P(y = j \left| \mathbf{x})$

10: Reject input as unknown if $y^* == 0$ or $P(y = y^* \left| \mathbf{x}) < \varepsilon$

4 Experimental Results

4.1 Classification without rejection

In this experiment, we divided all 6 categories in the dataset into training and test sets in an 80:20 ratio, while retaining the improved OpenMax method for estimating unknown classes. The confusion matrix is shown in Figure 2. The results compared to similar studies can be seen in Table 3.

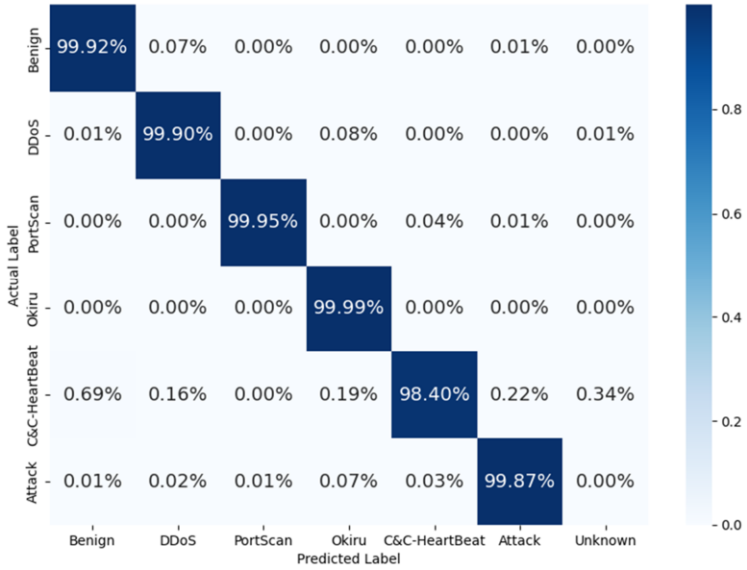

Figure 2. Confusion matrix of classification without rejection

Table 3. Comparison of similar studies

Method	Average accuracy	Precision	Recall	Weighted F1 Score
Random Forest [9]	93.35%	93.77%	93.12%	93.32%
PCA + SVM [10]	89.15%	87.58%	91.46%	90.31%
CNN [11]	96.76%	97.21%	92.39%	94.65%
FFN [12]	99.37%	99.08%	**99.71%**	99.39%
Propose Method	**99.80%**	**99.82%**	99.66%	**99.74%**

4.2 Performance on unknown samples rejection

In this experiment, we partitioned 80% of the top four categories (Benign, DDoS, Part of Horizontal Port Scan, Okiru) as the training set, while the remaining along with the two other categories (C&C-Heartbeat, Attack) was used as the test set in this experiment. The experimental results are shown in Figures 3 and 4. It can be seen that the model without EVT wrongly identifies unknown classes as known classes with high confidence. The right side applies Algorithm 1, which reduces the accuracy of known class classification to 95% on average, but can reject unknown samples with an accuracy of over 90%.

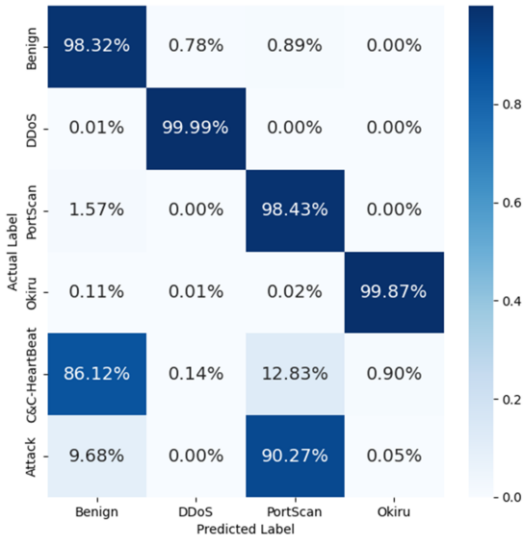

Figure 3. Classification without EVT

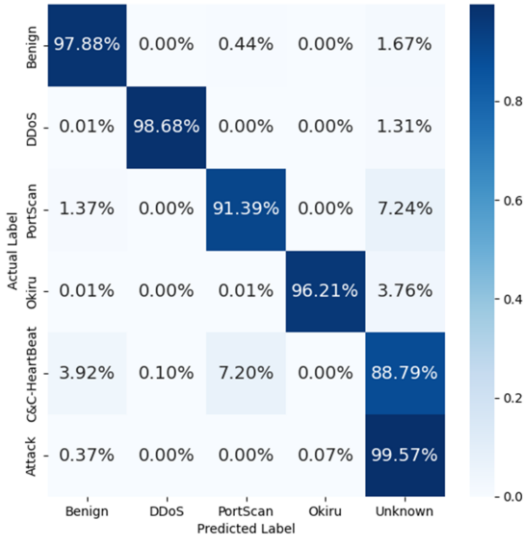

Figure 4. Classification with EVT

4.3 Ablation study on focal loss

In this subsection, we compare the average distance between all categories of MAV under different Loss functions in Table 4, which reflects the model's delineation of category boundaries. The average distance obtained by Focal Loss is 104% farther than cross entropy loss, which indicates that different categories are farther apart and have clearer boundaries under Focal Loss. We also simply selected 5000 random samples for t-SNE analysis after model prediction, and the results can be seen in Figures 5 and 6. In the t-SNE analysis, it can be observed that compared to Figure 5, the unknown classes labeled

as 4 (yellow) and 5 (green) in Figure 6 are more concentrated. Also, for known classes such as class 2, using Focal Loss can make the boundaries of known classes clearer.

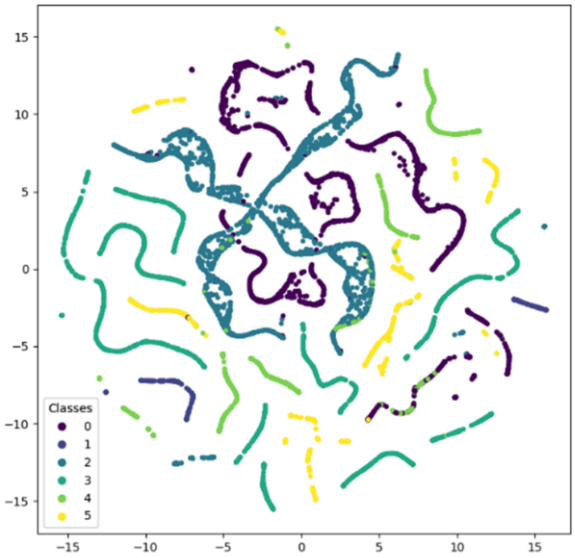

Figure 5. t-SNE result while using cross entropy loss

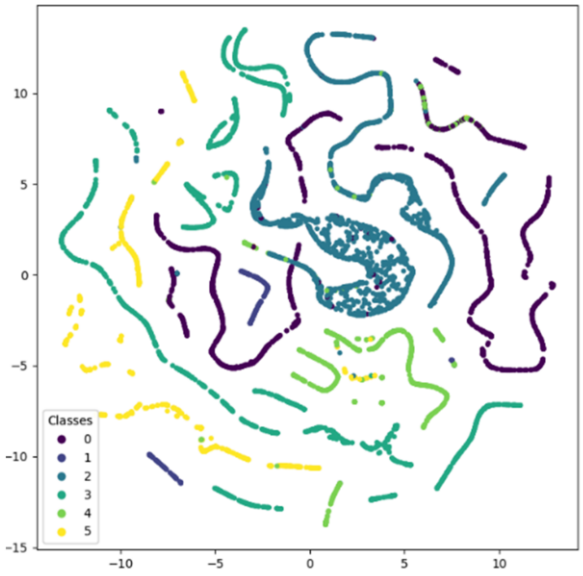

Figure 6. t-SNE result while using Focal Loss

Table 4. Comparison of distances with different loss functions

Loss function	Inter-class distance
Cross entropy loss	7.3249
Focal loss	14.9437

5 Conclusions

This research focused on developing a traffic classification model for the IoT23 dataset. The study introduced an improved OpenMax algorithm for rejecting unknown traffic, enhancing the model's real-world applicability. Experimental results demonstrated that combining the Focal Loss function with OpenMax improved the model's performance by increasing class separability.

Furthermore, the experimental evaluation highlighted the effectiveness of combining the Focal Loss function with OpenMax. By increasing class separability, this combination improved the overall performance of the traffic classification model. The findings contribute to the field by providing a more reliable and secure environment for IoT networks through accurate classification of known traffic and effective rejection of unknown traffic instances.

References

[1] Lippmann, R.P., et al. (2000) Evaluating intrusion detection systems: the 1998 DARPA off-line intrusion detection evaluation. In: DARPA Information Survivability Conference and Exposition. Hilton Head. pp. 12-26. DOI: 10.1109/DISCEX.2000.821506.

[2] Wang, Z. (2015) The applications of deep learning on traffic identification. BlackHat USA. 24: 1-10. https://blackhat.com/docs/us-15/materials/us-15-Wang-The-Applications-Of-Deep-Learning-On-Traffic-Identification-wp.pdf

[3] Wang, W., Zeng, X., Ye, X., Sheng, Y., Zhu, M. (2017) Malware Traffic Classification Using Convolutional Neural Networks for Representation Learning. In: 31st International Conference on Information Networking. Da Nang. pp. 712-717. DOI: 10.1109/ICOIN.2017.7899588

[4] Koroniotis, N., Moustafa, N., Sitnikova, E., Turnbull, B. (2019) Towards the development of realistic botnet dataset in the internet of things for network forensic analytics: Bot-iot dataset. Future Generation Computer Systems, 100: 779-796. DOI: 10.1016/j.future.2019.05.041

[5] Moustafa, N. (2021) A new distributed architecture for evaluating AI-based security systems at the edge: Network TON_IoT datasets. Sustainable Cities and Society, 72: 102994. DOI: 10.1016/j.scs.2021.102994

[6] Garcia, S., Parmisano, A., Erquiaga, J. (2020) IoT-23: A labelled dataset with malicious and benign IoT network traffic (Version 1.0.0). Zendo. DOI: 10.5281/zenodo.4743746

[7] Bendale, A., & Boult, T. E. (2016). Towards open set deep networks. In: IEEE conference on computer vision and pattern recognition. Las Vegas. pp. 1563-1572. DOI: 10.1109/CVPR.2016.173

[8] Lin, T. Y., Goyal, P., Girshick, R., He, K., Dollár, P. (2017). Focal loss for dense object detection. In: IEEE international conference on computer vision. Venice. pp. 2980-2988.DOI: 10.1109/ICCV.2017.324

[9] Ullah, I., Mahmoud, Q. H. (2021). Design and development of a deep learning-based model for anomaly detection in IoT networks. IEEE Access, 9: 103906-103926. DOI: 10.1109/ACCESS.2021.3094024

[10] Nanthiya, D., Keerthika, P., Gopal, S. B., Kayalvizhi, S. B., Raja, T., & Priya, R. S. (2021). SVM Based DDoS Attack Detection in IoT Using Iot-23 Botnet Dataset. In: 2021 Innovations i-PACT. Kuala Lumpur. pp. 1-7. DOI: 10.1109/i-PACT52855.2021.9696569.

[11] Strecker, S., Dave, R., Siddiqui, N., & Seliya, N. (2021). A modern analysis of aging machine learning based IOT cybersecurity methods. arXiv preprint. https://arxiv.org/abs/2110.07832.

[12] Ullah, I., Mahmoud, Q. H. (2022). An anomaly detection model for IoT networks based on flow and flag features using a feed-forward neural network. In: 2022 IEEE 19th CCNC. Las Vegas. pp. 363-368. DOI: 10.1109/CCNC49033.2022.9700597.