International Journal of Scientific Research in Computer Science, Engineering and Information Technology
ISSN : 2456-3307 (www.ijsrcseit.com)
doi : https://doi.org/10.32628/CSEIT206452

Outlier Detection in IoT Using Generative Adversarial Network

B. Joyce Beula Rani, Prof. L. Sumathi M.E.
Computer Science and Engineering, Government College of Technology, Coimbatore, Tamil Nadu, India

Article Info
Volume 6, Issue 4
Page Number: 306-311
Publication Issue :
July-August-2020

Article History
Accepted : 20 July 2020
Published : 30 July 2020

ABSTRACT

Usage of IoT products have been rapidly increased in past few years. The large number of insecure Internet of Things (IoT) devices with low computation power makes them an easy and attractive target for attackers seeking to compromise these devices and use them to create large-scale attacks. Detecting those attacks is a time consuming task and it needs to be identified shortly since it keeps on spreading. Various detection methods are used for detecting these attacks but attack mechanism keeps on evolving so a new detection approach must be introduced to detect their presence and thus blocking their spreading. In this paper a deep learning approach called GAN – Generative Adversarial Network can be used to detect this outlier and achieve 85% accuracy.

Keywords : IoT, Botnets, Deep Learning, GAN, Outlier

I. INTRODUCTION

The Internet of things (IoT) is the network of physical devices, vehicles, home appliances, and other items with embedded electronics, software's, sensors, actuators and connectivity which enables these things to connect, collect and exchange data. Compared to conventional computing systems, IoT systems are at higher security risk for several reasons. [1]

- IoT systems don't have well defined perimeters and continuously change due to device and user mobility.
- IoT systems are highly heterogeneous with respect to communication medium and protocols, platforms, and devices.
- IoT devices could be autonomous entities that control other IoT devices.
- IoT systems might include things not designed to be connected to the Internet.
- IoT systems, or portions of them, could be physically unprotected and/or controlled by different parties.
- Unlike smartphone applications, which require permission for installation and many user interactions, granular permission requests might not be possible in IoT systems because of the large number of devices.

Recently, deep learning approach have been utilized for solving many problems in the area of outlier detection include botnet detection.

The structure of the remaining paper is as follows: Section B describes the literature survey and Section C discusses the implementation and results Section D

provides the conclusion and suggestions for some future work.

II. LITERATURE SURVEY

The survey begins with the introduction of GAN methodology and limits discussion to the application of GAN in the field of outlier detection and various approaches in outlier detection in IoT environment.

A GAN framework was first proposed in [3] that simultaneously train two models. Generative model G, captures the data distribution and Discriminative model D, estimate the probability that a sample came from the training data rather than G. Both networks were defined by Multilayer Perceptron and the entire system can be trained with Back propagation. Three datasets MNIST, TFD, CIFAR-10 were used. Minibatch stochastic gradient descent algorithm was implemented in the training of GAN. Measure the probability of the test set data by fitting a Gaussian Parzen window.

A Conditional GAN was proposed in [4], that jointly learns the generation of high dimensional image space and the inference of latent space. This model consists of two encoders, decoder and a discriminator. Three types of dataset were used MNIST, CIFAR and X-ray security screening. For comparison of the model on the three datasets, AUC and Runtime were implemented. This method achieves highest runtime and attains highest AUC.

A BiGAN (Bidirectional GAN) was developed in [5] that simultaneously learn an encoder E which maps input samples x to a latent representation z. This avoids the computationally expensive step of recovering a latent representation at test time. Two datasets MNIST and KDD99 were used. Highest anomaly scores A(x) were measured and classified as anomalies.

Adversarial Learned Inference (ALI) within the GAN framework [6] was proposed which jointly learns a generator network and an inference network model consists of encoder, decoder and discriminator. Four datasets SVHN, CIFAR-10, CelebA and ImageNet were used. In CelebA, it attains lower misclassification rate of 3.00 ± 0.50%, SVHN attains 7.42 ± 0.65, CelebA attains 17.05 ± 1.49.

A Rand Net (Randomized Neural Network) was proposed in [7] to perform unsupervised outlier detection with auto encoder ensembles based on reconstruction error rate. Auto encoders with ensemble framework were randomly connected to achieve improved diversity and training time. To speed-up the training process, adaptive sampling methods were used and achieve major gain.

Entropy based K-NN model was proposed in [8] to perform outlier detection in Wireless Sensor Networks by vary the distance types and the number of nearest neighbor ' K ' value. Real time dataset was used and achieve highest accuracy of 86 % with K value as 2 at secludian distance approach.

EDIMA approaches various machine learning techniques, [9] were used to detect the IoT malware network activity in large scale networks based on their scanning traffic patterns. Three classification algorithms Random Forest, K-NN and Gaussian Naïve Bayes approaches were implemented. K-NN performs best with high accuracy, precision, recall and F1 score.

Hidden Markov Model (HMM) approach was implemented [10] to identify anomalous activities that can occur in a smart home environment. Two types of data were collected, network data and behavioral data and detect the abnormal behavior. This approach achieves 97 % of accuracy.

NTAD – Network Traffic Anomaly Detection approach in [11] with two levels of PCA techniques were implemented. Few principal components were used for quick detection in the first level and another few components were used for detailed detection. Distance calculation methods were implemented and provide acceptable TPR and FPR.

Decision Tree classifier is used [12] to detect the IoT Bots with less number of features in a multiclass supervised setting. Fishers score was used for feature selection to minimize the number of features in detecting the IoT bots. N-BaIoT dataset is used and achieve high accuracy detection of 98%.

TABLE 1. COMPARATIVE STUDY OF GENERATIVE MODELS

Paper	Issue	Technique used	Dataset	Metrics
Generative Adversarial Nets[3]	Require markov chains , inference rules.	GAN framework	MNIST, CIFAR	High accuracy parzen window
Conditional Generative Adversarial Nets[4]	Scaling, mapping	Conditional GAN	MNIST, MIR FLICKR	High accuracy Parzen window
Efficient GAN-based Anomaly Detection[5]	Complex & high dimensional data distribution , computational expensive	BiGAN	Image, KDD99	Above 90% of Precision, Recall
Adversarially Learned Inference[6]	Complexity in learning the inference	ALI model	CIFAR10, SVHN, CelebA and ImageNet	Achieve lower misclassifi cation rate.
Outlier Detection with Autoencoder Ensembles[7]	Sensitive to noise, require large dataset.	RandNet (AE+adaptive sampling method)	Real datasets	Achieve accuracy Outlier score
Dimensionality Reduction for Machine Learning Based IoT Botnet Detection[12]	Severe security problems, Huge Botnets	Decision tree classifier with dimensionality reduction	N-BaIoT dataset	Achieve 98 % accuracy

III. IMPLEMENTATION

For Experiment, GAN framework is developed in Keras with TensorFlow as backend and experiments were conducted using the IoT botnet dataset.

(i) DATA

In this environment different IoT devices were connected to an isolated network using both wired and Wi-Fi connections. Also botnet components were installed in the network, such as C&C server. By using port mirroring at the internet switch, data was gathered using wireshark for both normal traffic, and malicious traffic.

(ii) DATASET

The dataset utilized in our study includes the statistics of network traffic captured in a lab environment in which the typical normal behavior and attack cases are simulated [2].The dataset contains data from 9 IoT devices such as smart-home and security domain and security domains such as doorbells, security cameras, thermostat, baby monitor etc. The malicious traffic includes the attacks by Bashlite and Mirai malware.

In total data consists of 115 features which are related to different information about packets , aggregated in five categories .Source Host IP, Source MAC-IP, Source and destination host IP, Source and destination host and port, Source and destination host traffic jitter. Each statistics were calculated for different window sizes: 100ms, 500ms, 1.5sec, 10sec, and 1min using lambda decay parameter. The network statistics data consists of weight or packet count; mean of packet size; variance of packet size; standard deviation of packet size; radius as root squared sum of two stream's variances; magnitude of two stream's means; covariance and Pearson's correlation coefficient. Packet counts, mean and variance of packet sizes, are included in all categories. Additionally, more detailed statistics such as magnitude, radius, covariance and correlation coefficient of packet sizes are given for channel and socket categories.

TABLE 2. Feature Categories

Feature Categories	Features
Host-IP	Packet count, mean and variance (outbound)
Host-MAC&IP	Packet count, mean and variance (outbound)
Channel	Packet count, mean and variance (outbound) Magnitude, Radius, Covariance,Correlation Coef. (inbound and outbound)
Network Jitter	Count, mean and variance of packet jitter in channel
Socket	Packet count, mean and variance (outbound) Magnitude, Radius, Covariance, Correlation Coefficient (inbound and outbound)

Further, Mirai data consists of 5 attack classes: SYN, ACK, UDP, UDPPlain, SCAN. In total dataset consists of more than 5 million points.

(iii) GAN FRAMEWORK

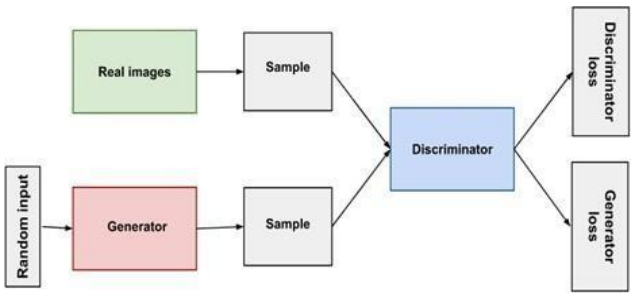

FIGURE: 1 OVERVIEW OF GAN STRUCTURE

The architecture of a GAN has two basic elements, the generator network and the discriminator network. Each network can be any neural network such as an Artificial Neural Network (ANN), a Convolutional Neural Network (CNN), a Recurrent Neural Network (RNN) or a Long Short Term Memory (LSTM). The discriminator has to have fully connected layers with a Classifier at the end. Both the generator and the discriminator are neural networks. The generator output is connected directly to the discriminator input. Through back propagation, the discriminator's classification provides a signal that the generator uses to update its weights. Through multiple cycles of generation and discrimination, both networks train each other, while simultaneously trying to outwit each other.

(iv) TRAINING PHASE

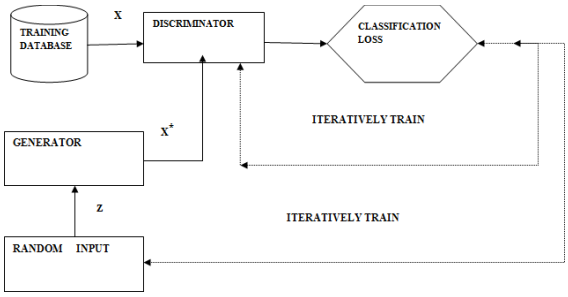

FIGURE : 2 TRAINING A GAN

TRAINING PROCESS

The training process follows a framework that corresponds to the min-max two-player game. In this game discriminator try to discriminate the image generated by generator from actual training sample and generator try to fool discriminator into believing that image indeed has come from training sample. Training stops when discriminator predicts a probability of 0.5 for each output generated by the generator. We can say that we have reached a stage where the generator has almost captured the distribution of training samples and can independently convert random data to look like the sample has come from training data. Both discriminator and generator can be declared multilayer perceptron and the whole system can be trained using backpropagation. The training process for both generator and discriminator happens simultaneously. We train discriminator during each epoch of total data to get maximum accuracy of predicting correct label whereas generator is trained in parallel to minimize its loss function by generating as accurate image as possible which can confuse discriminator in mislabelling it.

Network based IoT dataset is used to train both generator and discriminator network. Both are sequential model with dense and fully connected layer of CNN. Generator takes z as input with noise dimension as 100 and produces the data of 115

features. Leaky ReLu activation function is used at each layer. Discriminator classifies both real and fake data and dropout function is used to downsample. The loss is calculated for each of these models, and the gradients are used to update the generator and discriminator.

TensorFlow is an interface to express machine learning algorithms. The computation using TensorFlow can be expressed with little or no change in its state on a wide variety of systems like distributed systems and computational devices. The TensorFlow packages were released as an open-source package under license of the Apache 2.0 in November, 2015 and are available at www.tensorflow.org.

IV. RESULTS

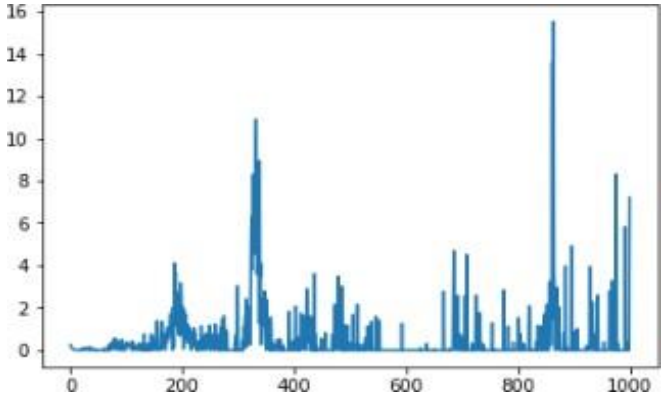

FIGURE: 3 GENERATOR DATA

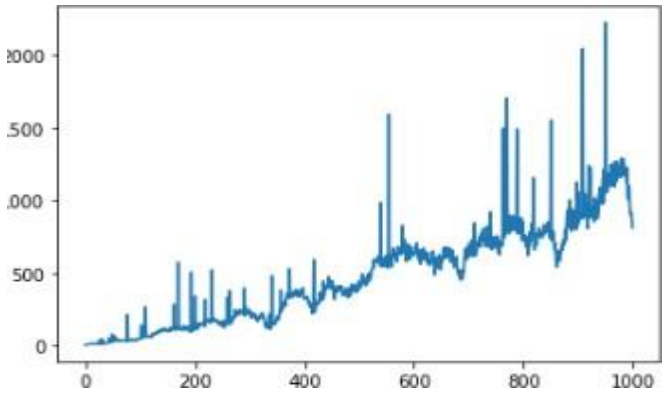

FIGURE: 4 DISCRIMINATOR DATA

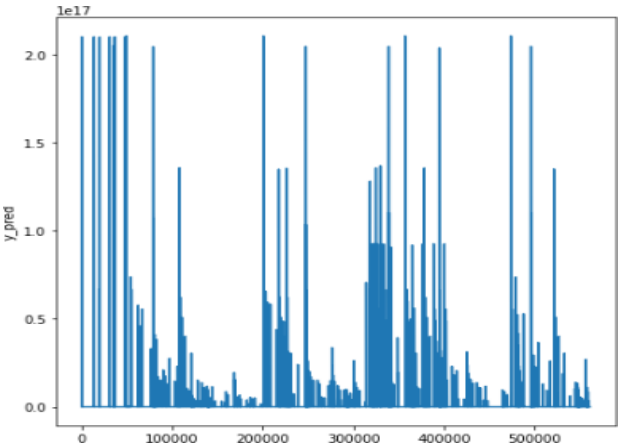

FIGURE: 5 PREDICTION DATA

V. CONCLUSION

The best threshold value is prob = 0; While GAN is widely used in computer vision problem, it does perform very well in tabular data. One of the most common problem in anomaly detection is class imbalance. While using GAN, we indirectly overcome the problem as we only training our model with just one of the classes. Besides, we only training our model with just 500 data which is a very small amount when comparing to the data required by other ML/DL models.

VI. REFERENCES

[1]. C. Kolias, G. Kambourakis, A. Stavrou, and J. Voas, (2017). DDoS in the IoT: Mirai and Other Botnets. Computer, vol. 50, pp. 80–84.

[2]. Yair Meidan, Michael Bohadana, Yael Mathov, Yisroel Mirsky, Dominik Breitenbacher, Asaf Shabtai, and Yuval Elovici, (2018). N-BaIoT: Network-based Detection of IoT Botnet Attacks Using Deep Autoencoders. Pervasive Computing , vol.13 , pp.83-90.

[3]. Goodfellow, Ian, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, SherjilOzair, Aaron Courville, and YoshuaBengi, (2014) Generative adversarial nets. In Advances in neural information processing systems, pp. 2672-2680.

[4]. Akcay, Samet, Amir Atapour- Abarghouei, and Toby P. Breckon, (2018). Ganomaly: Semi-supervised anomaly detection via adversarial training. In Asian Conference on Computer Vision, Springer, Cham. pp. 622-637.

[5]. Zenati, Houssam, Chuan Sheng Foo, Bruno Lecouat, GauravManek, and Vijay Ramaseshan Chandrasekhar, (2018). Efficient gan- based anomaly detection. arXiv preprint arXiv: 1802.06222 .

[6]. Dumoulin, Vincent, Ishmael Belghazi, Ben Poole, Olivier Mastropietro, Alex Lamb, Martin Arjovsky, and Aaron Courville, (2016). Adversarially learned inference. arXiv preprint arXiv: 1606.00704 .

[7]. Chen, Jinghui, SaketSathe, CharuAggarwal, and Deepak Turaga, (2017). Outlier detection with autoencoder ensembles. In Proceedings of the 2017 SIAM International Conference on Data Mining, Society for Industrial and Applied Mathematics, pp. 90-98.

[8]. Yadav, Manmohan Singh and Shish Ahamad, (2018). Anomaly Detection in Wireless Sensor Networks-Critical Survey .

[9]. Kumar, Ayush and Teng Joon Lim, (2019). EDIMA: Early Detection of IoT Malware Network Activity Using Machine Learning Techniques. IEEE 5th World Forum on Internet of Things (WF-IoT) (2019): 289-294.

[10]. Ramapatruni, Sowmya, Sandeep Nair Narayanan, Sudip Mittal, Anupam Joshi and Karuna Pande Joshi, (2019). Anomaly Detection Models for Smart Home Security. IEEE 5th Intl Conference on Big Data Security on Cloud (Big Data Security), IEEE Intl Conference on High Performance and Smart Computing, (HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS) : 19-24.

[11]. Hoang, Dang Hai and Ha Duong Nguyen, (2019). Detecting Anomalous Network Traffic in IoT Networks. 21st International Conference on Advanced Communication echnology (ICACT) : 1143-1152.

[12]. Bahsi, Hayretdin, Sven Nomm and Fabio BenedettoLa Torre, (2018). Dimensionality Reduction for Machine Learning Based IoT Botnet Detection. 15th International Conference on Control, Automation, Robotics and Vision (ICARCV) :1857-1862.

[13]. E. Bertino and N. Islam, (2017). Botnets and Internet of Things Security. Computer, vol. 50, pp.162-166.

[14]. Nidhi Srivastav and Rama Krishna Challa, (2013). Novel intrusion detection system integrating layered framework with neural network. Advance Computing Conference (IACC), IEEE 3rd International, pages 682–689.

Cite this article as :

B. Joyce Beula Rani, Prof. L. Sumathi, "Outlier Detection in IoT Using Generative Adversarial Network", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 6 Issue 4, pp. 306-311, July-August 2020. Available at doi : https://doi.org/10.32628/CSEIT206452
Journal URL : http://ijsrcseit.com/CSEIT206452