ISSN 2321 3361 © 2021 IJESC

Research Article Volume 11 Issue No.08

Botnet Detection in Network Traffic Based on GBM

Somesh Kumar Sahu[1], Kiran Muloor[2], Shashidhara G. M[3], Sandeep Shyam Bajaj[4]
REVA Academy for Corporate Excellence, REVA University, Bangalore, India

Abstract:
Over the past decade botnets have gained the attention of many security teams in the companies and researchers across the globe. Security teams are working tirelessly to develop systems that would detect the botnet with high accuracy in the network traffic. Botnet attacks are unique threats to systems and has high vulnerability these high risks problems naturally attracted researchers and professionals and started applying machine learning (ML) techniquesto detect botnet attacks. We would like to evaluate different features and the result impact on detection accuracy for a given machine learning method used.We understand that the network traffic is being analyzed through various classification machine learning models and has given good results but we have not come across any research paper or could be less work done on Gradient Boosting Machine (GBM). We see a scope to work on GBM detecting botnet and hence propose GBM algorithm to classify the botnet traffic. In this paper, we focused only on the preprocessed botnet classified data.

Keywords: Anomaly detection, Botnet, Gradient Boosting Machine, AUC,.

I. INTRODUCTION

Botnets [4] have been for a long time one of the primary security threats on the Internet and the threat risk is increasing day by day with new types of Botnet penetrating security layers. It is easy to infuse a botnet in to any network, hackers today are equipped to attack quickly after exploiting new vulnerabilities. Many Machines numbering in thousands are typically part of a single botnet. Botnets are highly dynamic in nature and hard to detect because of theadapting behavior and can easily breach the most common security defenses. Botnet attacks will arise from different sources and will be of different types of adaptability will be a challenge in any model or Machine learning algorithm developed. This can be achieved by using machine learning techniques mentioned in the paper. In today's world numerous techniques exist to identify a specific kind of botnet (Telnet, IRC, P2P, Domains, etc. [7]) and the scope is very much limited to a few. Limiting the application of the model in the current IoT age. To overcome the limitations, we will be targeting on differentiating normal network traffic.

II. LITERATURE REVIEW

A study by Hoang et.al (2018) shows the botnet detection model built using machine learning techniques based on Domain Name System query data. The output on Domain Generation Algorithms botnet and Fast FluX botnet data demonstrate that most of the machine learning techniques is classification- models with an accuracy of 85%. Among all the models, the random forest algorithm has given the good results with an accuracy of 90.80%. [1] Wai, F. K. et.al (2018) used a data set which used to train a binomial classifier to detect anomaly in input traffic. The study has used techniques such as linear Support Vector Machines, Decision Tree (DT) and Random Forests (RF). Support Vector Machines builds the optimum linear hyperplane and classifies the data into two classes. RF produces lowest False Positive Rate, Decision Treeresults in higher Recall and FPR. The conclusion from this study is that the DT training data doesn't have overfitting issue when compared to RF model. The results indicate that Decision Tree model turned out to be the best classifier [2]. Khan, R. U et.al (2019) analyzed most common attacks like SPAM, Port Scan, Fast Flux, IRC, Click Fraud, DDoS, Compiled and Controlled record by CTU, HTTP, Waledac, Storm and Zeus botnets. From the study it was observed that the Decision Tree algorithm had a high accuracy to detect P2P botnet traffic. [3]. Wei et al. 2016 [12] usedclustering method instead of classification, which is an unsupervised machine learning technique.The work was to analyze similarity analysis malicious and benign data. (4).However, the study had very limited scope of identifying botnet from one particular host.

III. OBJECTIVE OF THE STUDY

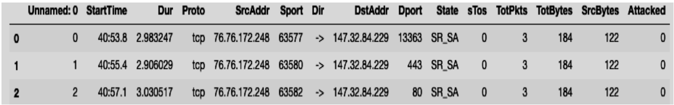
This paper applies the Cross-Industry Standard Process for Data Mining(CRISP-DM) is a complete data mining method and a structure that gives complete view of conducting data mining in this paper. CRISP-DM has different phases in the life cycle of a data mining[22] and we have followed all those phases and explained below.

IV. BUSINESS UNDERSTANDING

IT security team's Business objective is to prevent company devices from becoming part of a botnet and protect corporate assets from botnet attacks. Using Analytics the use case is to comprehend the pattern of botnet attacks and help IT, security team, to take appropriate measures in preventing malicious attacks on business. The data on number of attacks has increased by 84%, and the DDoS attacks has doubled. The average duration of the impact increased by 4.21 times, while the extremely long attacks incraseda massive 487% growth.[21] By 2022 there will be 1.5 mobile devices per capita. There will be 12.3 billion mobile-connected devices by 2022.[20] These data points show that the Botnet threat is increasing year by year and every second number of devices are getting added to the network The objective of this study is to label the Network traffic and predict Normal and Botnet traffic accurately.

V. DATA UNDERSTANDING

We have accessed online files with preprocessed data [11] this dataset has been labelled with normal, botnet attack and background traffic. It has 16 features and 71 different types of botnet samples. Files were in pcap format and we converted them into CSV format. Sample data that we have accessed which has labels and have different attributes



	Unnamed: 0	StartTime	Dur	Proto	SrcAddr	Sport	Dir	DstAddr	Dport	State	sTos	TotPkts	TotBytes	SrcBytes	Attacked
0	0	40:53.8	2.983247	tcp	76.76.172.248	63577	->	147.32.84.229	13363	SR_SA	0	3	184	122	0
1	1	40:55.4	2.906029	tcp	76.76.172.248	63580	->	147.32.84.229	443	SR_SA	0	3	184	122	0
2	2	40:57.1	3.030517	tcp	76.76.172.248	63582	->	147.32.84.229	80	SR_SA	0	3	184	122	0

Figure.1. Sample data

Pre-processed data considered for this experiment has 2% anomalies across the distribution of network traffic. The data is imbalanced with only 2% of anomalies this hinders the learning performance of learning models. Neglecting the unbalance data leads to negative consequences [12]. The underlying concept of supervised learning technique is that it requires appropriate data for learning. Classifiers are modeled to learn and predict. Sufficient data required for training the model otherwise it would hinder the Learning.

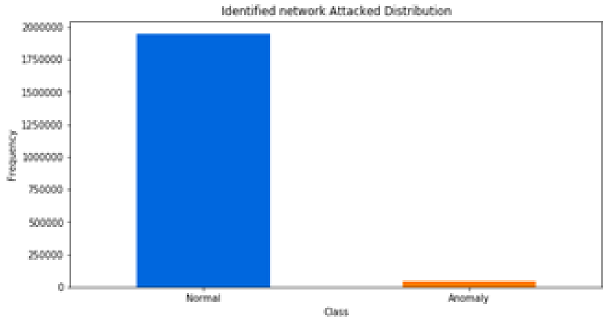


Figure.2. Actual data with 2% anomaly

VI. DATA PREPARATION

The objective of any Machine learning (ML) algorithms is minimizing errors. “Since the probability of instances belonging to the majority class is significantly high in the imbalanced data set, the algorithms are much more likely to classify new observations to the majority class” [13].We have used the up-sampling technique to increase minority (Anomaly) data so that we can overcome the problem of overfitting of the model as the anomaly rate is only 2%[15] In this case we have up-sampled the Minority -class to 55 % of Majority class.

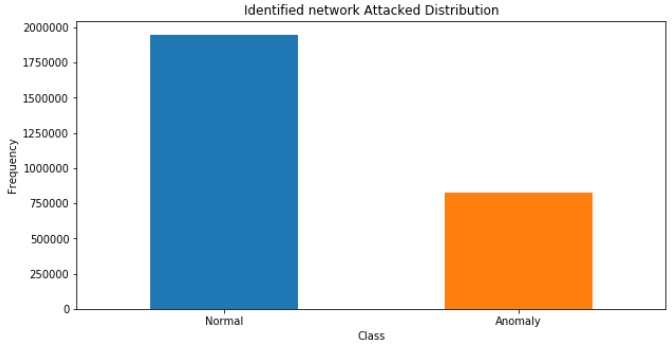


Figure.3. After up-sampling data with 55% anomaly

VII. MODELING

The botnet prediction model starts with a detailed analysis by investigating botnet features and behavioral attributes. This analysis can be generated through different classification techniques [6]. Using classifier technique, we can understand the features and behavior of Botnet. From the dataset split data for training and testing purposes. Extract the features of this dataset, we bifurcated these features into two classes, Attacked and Not Attacked. Machine learning studies how to automatically discover to make accurate predictions based on past observations. We have applied GBM (Gradient Boost Machine).

A. Classification Technique

Classification is a process of extracting the features of given data set and find out the behaviour of Botnet and its patterns for this purpose different classifier techniques are used [8]. It gives a thorough study of network traffic and determining Botnet, detects and analyzes the evidence accuracy using an efficient machine learning algorithm. The classifying process is adopted when confirmed that there is a bot in a network and it is already identified through our experimentation.Our analysis result shows the performance for findings Bot evidence using theclassifiers with more detected accuracy [9]. We opted H2o GBM classifier for detecting malicious bots with higher accuracy.

Gradient Boosting Machine(GBM)

“GBM (Regression and Classification) is anwidely used forward ML ensemble method.[14] that has demonstrated over many domains. The guideline behind GBM is to construct on weak successive trees with each tree learning and progressing on the past ones. GBM sequentially builds regression trees on all the variables of the dataset in complete each tree is built in parallel.
Gradient Boosting converts decision trees as weak learners into strong learners. In boosting, trees are added without changing the existing trees.When a new tree is fitin GBM it will be done on modified version while retaining the original data set. In Gradient Boosting algorithm (GBM) algorithm starts by training a decision tree and each observation is given an equal weight. After first tree evaluation, the weights are changed either by increasing the weights of those observations that are not classified or decrease the weights of those observations that are easy to classify. Continuing the process on the reference of weighted data, the second tree is grown this improves better predictions based on the intialtree. [16] Once the new model is built with Tree 1 and Tree 2 we will derive the error from Tree 2 and build ensemble model and begin building new third tree to predict the new error. The GBM trains many models gradually and the process is additive and continuous in sequential manner. This is an iterative process subsequent trees are grown to easily classify data that were not classified correctly by the previous trees. The final ensemble model predictions are made using the previoustree models predicted weighted sum. “The GBM can also be tuned to get optimal combination of hyperparameters, Common parameters are Number of trees, depth of trees and learning rate”.[5] The loss function is a measure which indicates how good the models are at fitting the original data.[18], The GBM emphasize on MSE where it outlines the approach of additive and sequentially fitting the trees to minimize the error. In this paper GBM H2O is used for the experiments. H2O is an open-source, in-memory, fast, and scalable machine learning helps in predictive analytics platform.[24]

VIII. RESULTS EVALUATION AND DISCUSSION

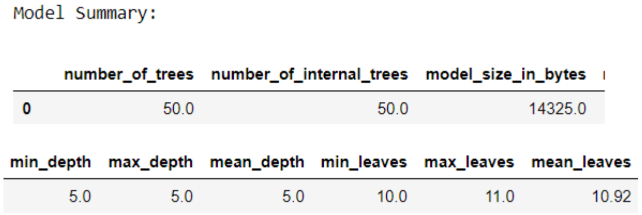
Experiments were conducted using Botnet attack records. The Botnet data contains 1997072 records out of which 40K were identified as botnet attack records which is around 2%. The current paper deals with binary classification problem, two classes (Normal, Attacked), Up-sampling applied to reduce the imbalance. In this paper, we will be examining key metrics of GBM.

Key Metrics:
a)"MSE: It is the average squared difference between the predicted and actual target variables, and its best value is 0.0"[18]
b)"AUC: classification model distinguishes true positives and false positives. AUC of 1 is a perfect model classifier; an AUC of 0.5 indicates a poor classifier of random guessing". [18]
c) "Accuracy: The model accuracy is determined by the number of correct predictions made to the ratio of all possible predictions made."[18]
d) "Cross – Validation: Cross-validation is to trim training set into k blocks, then use one of those k blocks as a validation data set, and use the rest for training. Repeat this k times, with a different part of the training set being the validation set each time".[19].

GBM Results

We have split the data into different Split of Train,Test and Valid[17] percentages.

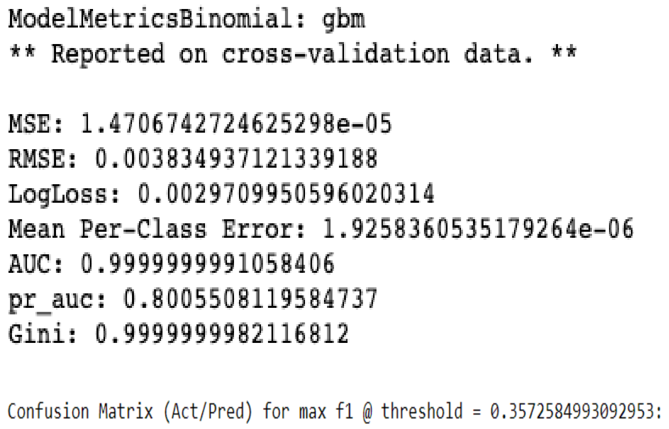
1. Experiment 1 :Train (80%) Valid (20%)



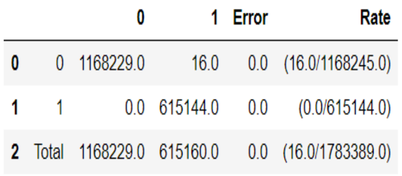
```
Model Summary:
```

	number_of_trees	number_of_internal_trees	model_size_in_bytes
0	50.0	50.0	14325.0

min_depth	max_depth	mean_depth	min_leaves	max_leaves	mean_leaves
5.0	5.0	5.0	10.0	11.0	10.92

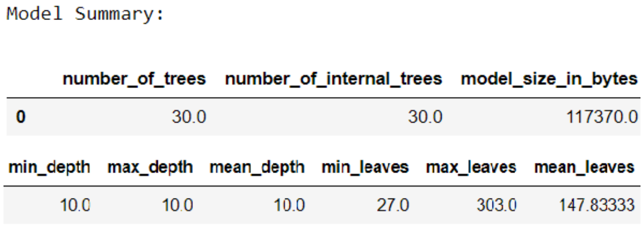


```
ModelMetricsBinomial: gbm
** Reported on cross-validation data. **

MSE: 1.4706742724625298e-05
RMSE: 0.003834937121339188
LogLoss: 0.002970995059602031 4
Mean Per-Class Error: 1.9258360535179264e-06
AUC: 0.9999999991058406
pr_auc: 0.8005508119584737
Gini: 0.9999999982116812

Confusion Matrix (Act/Pred) for max f1 @ threshold = 0.3572584993092953:
```



		0	1	Error	Rate
0	0	1168229.0	16.0	0.0	(16.0/1168245.0)
1	1	0.0	615144.0	0.0	(0.0/615144.0)
2	Total	1168229.0	615160.0	0.0	(16.0/1783389.0)

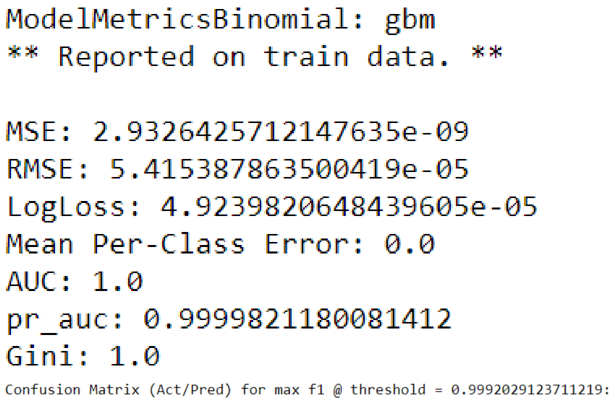
2. Experiment 2: Train (70%) Test(15%) Valid(15%)



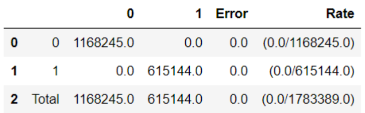
```
Model Summary:
```

	number_of_trees	number_of_internal_trees	model_size_in_bytes
0	30.0	30.0	117370.0

min_depth	max_depth	mean_depth	min_leaves	max_leaves	mean_leaves
10.0	10.0	10.0	27.0	303.0	147.83333

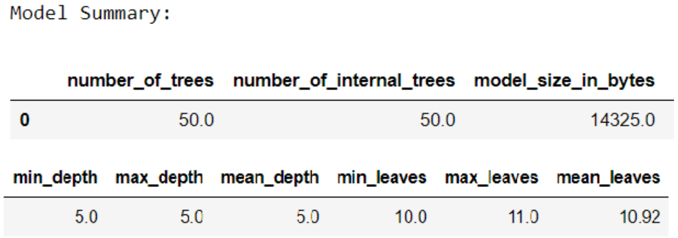


```
ModelMetricsBinomial: gbm
** Reported on train data. **

MSE: 2.9326425712147635e-09
RMSE: 5.415387863500419e-05
LogLoss: 4.9239820648439605e-05
Mean Per-Class Error: 0.0
AUC: 1.0
pr_auc: 0.9999821180081412
Gini: 1.0
Confusion Matrix (Act/Pred) for max f1 @ threshold = 0.9992029123711219:
```



		0	1	Error	Rate
0	0	1168245.0	0.0	0.0	(0.0/1168245.0)
1	1	0.0	615144.0	0.0	(0.0/615144.0)
2	Total	1168245.0	615144.0	0.0	(0.0/1783389.0)

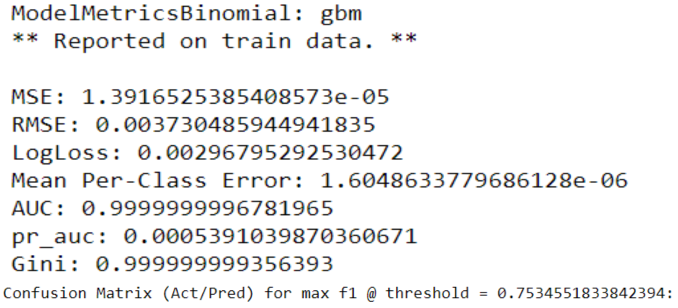
3. Experiment 3 :Train (60%) Test(20%) Valid(20%)



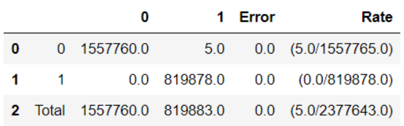
```
Model Summary:
```

	number_of_trees	number_of_internal_trees	model_size_in_bytes
0	50.0	50.0	14325.0

min_depth	max_depth	mean_depth	min_leaves	max_leaves	mean_leaves
5.0	5.0	5.0	10.0	11.0	10.92



```
ModelMetricsBinomial: gbm
** Reported on train data. **

MSE: 1.3916525385408573e-05
RMSE: 0.003730485944941835
LogLoss: 0.00296795292530472
Mean Per-Class Error: 1.6048633779686128e-06
AUC: 0.9999999996781965
pr_auc: 0.0005391039870360671
Gini: 0.999999999356393
Confusion Matrix (Act/Pred) for max f1 @ threshold = 0.7534551833842394:
```



		0	1	Error	Rate
0	0	1557760.0	5.0	0.0	(5.0/1557765.0)
1	1	0.0	819878.0	0.0	(0.0/819878.0)
2	Total	1557760.0	819883.0	0.0	(5.0/2377643.0)

GBM experiments on Botnet detection have given good performance with high accuracy and good botnet detection rate we have discussed two important metrics to substantiate the results.

AUC
We have carried three experiments by varying GBM hyper-parameters. The Accuracy in all three experiments is above 0.999 and in experiment 2 we have achieved AUC 1.0, higher the AUC the prediction is more accurate ranging from 0 to 1 [19]

Confusion Matrix
With different GBM hyper-parameters in all the three models the True Positives and True Negatives have been precisely classified with small error in Experiment 1 and 3. The precise classification of True positives and True negatives also mirrors with high AUC.

IX. CONCLUSION

The botnet has been a major threat to security and impacting business. The approach used in this study helps Cybersecurity teams to detect Botnet attacks proactively, increase network uptime and minimize the business impact. Based on the results using H2O GBM demonstrates high AUC ranging from 0.9999 to 1.0 and among the experiments we recommend using Experiment 2. Further we would like to expand our study in detecting different types of Botnet attacks.

X. REFERENCES

[1]. Hoang, X.D.; Nguyen, Q.C. Botnet Detection Based On Machine Learning Techniques Using DNS Query Data. Future Internet 2018, 10, 43.

[2]. Wai, F. K., Lilei, Z., Wai, W. K., Le, S., & Thing, V. L. (2018, October). Automated Botnet Traffic Detection via Machine Learning. In TENCON 2018-2018 IEEE Region 10 Conference (pp. 0038-0043). IEEE.

[3]. Khan, R. U., Zhang, X., Kumar, R., Sharif, A., Golilarz, N. A., &Alazab, M. (2019). An Adaptive Multi-Layer Botnet Detection Technique Using Machine Learning Classifiers. Applied Sciences, 9(11), 2375.

[4].Botnets: The New Threat Landscape -CISCO – Accessed on August 2019 http:// www. webtorials. com/ main/ resource/ papers/ cisco/paper99/botnets.pdf?source=post_page

[5]. http://uc-r.github.io/gbm_regression.

[6].Morstatter F, Wu L, Nazer TH, Carley KM, Liu H. A new approach to bot detection: striking the balance between precision and recall. InAdvances in Social Networks Analysis and Mining (ASONAM), 2016 IEEE/ACM International Conference on 2016 Aug 18 (pp. 533-540). IEEE.

[7].A. Bijalwan, N. Chand, E. S. Pilli, and C. R. Krishna, "Botnet analysis using ensemble classifier," Perspectives in Science, vol. 8, pp. 502-504, 2016.

[8].Zhao, D., Traore, I., Sayed, B., Lu, W., Saad, S., Ghorbani, A. and Garant, D., 2013. Botnet detection based on traffic behavior

[9].International Journal of Computer Science and Information Security (IJCSIS), Vol. 15, No. 7, July 2017

[10]. lass Logistic. Retrieved from http://weka. sourceforge. net/ doc.dev/weka/classifiers/functions/Logistic.html

[11]. Nitesh V. Chawla1, Aleksandar Lazarevic2, Lawrence O. Hall3, Kevin Bowyer4,

[12].GiovannaMenardi•Nicolahttp://cloud.politala.ac.id/politala/Jurnal/JurnalTI/Jurnal%2035/2Fs10618-012-0295-5.pdf

[13].https://medium.com/james-blogs/handling-imbalanced-data-in-classification-problems-7de598c1059f

[14].http://docs.h2o.ai/h2o/latest-stable/h2o-docs/data-science/ gbm.html

[15]. https:// medium.com/ bluekiri/ dealing- with -highly-imbalanced -classes-7e36330250bc

[16].https://www.kdnuggets.com/2019/02/understanding-gradient- boosting-machines.html

[17].https://www.oreilly.com/ library/ view/ practical -machine-learning/9781491964590/ch04.html

[18].http:// docs.h 2o.ai /h2o/ latest-stable/h2o-docs/_ sources/ performance-and-prediction.rst.txt

[19].https://derangedphysiology.com/main/cicm-primary-exam/ required-reading/research-methods-and-statistics/Chapter% 203.0.5/receiver-operating-characteristic-roc-curve

[20].https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/white-paper-c11-738429. html#_Toc953325

[21].https://securelist.com/ddos-report-q1-2019/90792/

[22].https://pdfs.semanticscholar.org/48b9/293cfd4297f855867ca278f7069abc6a9c24.pdf?_ga=2.213073492.1197755289.1572675415-1598846283.1565540135