第 48 卷 第 6 期
2021 年 6 月

湖 南 大 学 学 报（自 然 科学版）
Journal of Hunan University（Natural Sciences）

Vol. 48. No. 6.
June 2021

Open Access Article

Optimal Machine Learning Models for Kitsune to Detect Mirai Botnet Malware Attack

Abdullah Alabdulatif [1*], Syed Sajjad Hussain Rizvi [2], Manzoor Ahmed Hashmani [3]

[1] Computer Department, College of Science and Arts in Ar Rass, Qassim University, Ar Rass, Saudi Arabia

[2] Shaheed Zulfikar Ali Bhutto Institute of Science and Technology, Karachi, Pakistan

[3] Universiti Teknologi PETRONS, Seri Iskandar, Malaysia

Abstract: The network intrusion detection system (NIDS) is the key player to detect and mitigate Botnet Malware attacks. A plug-and-play NIDS, Kitsune, was proposed in the literature in 2018 as one of the best candidates. Kitsune's core algorithm is KitNET based on the ensemble of artificial neural networks called 'autoencoder' to classify legitimate and suspicious network traffic. Moreover, the Kitsune Network Attack dataset was donated to the UCI machine learning repository in October 2019. The study of Kitsune is found to be deficient in discussing the performance of other machine learning algorithms for Mirai Botnet malware attack detection besides artificial neural networks. Moreover, the study reported the performance as a true positive rate (TPR) and false-negative rate (FNR) only. In this paper, we propose that the selection of the model should be a function of TPR, FNR, training accuracy, test accuracy, misclassification cost, prediction speed, and train time. This paper presents a comprehensive investigation for selecting optimal machine learning model(s) for Kitsune. In this investigation, a large set of machine learning algorithms have opted. Our study reveals that the variants of tree algorithms such as Simple Tree, Medium Tree, Coarse Tree, RUSBoosted, and Bagged Tree have reported similar effectiveness but with slight variation inefficiency. Finally, Coarse Tree has won the competition and best-suited algorithm for Mirai botnet malware attack detection.

Keywords: cybersecurity, malware, botnet attack, Kitsune, network intrusion detection.

风筝 检测未来 僵尸网络恶意软件攻击的最佳机器学习模型

摘要：网络入侵检测系统是检测和缓解僵尸网络恶意软件攻击的关键角色。即插即用的网络入侵检测系统风筝于2018年在文献中被提出作为最佳候选系统之一。风筝的核心算法是风筝网，它基于称为“自动编码器”的人工神经网络集合，用于对合法和可疑的网络流量进行分类。此外，风筝网络攻击数据集于2019年10月捐赠给了加州大学尔湾分校机器学习存储库。发现风筝的研究缺乏讨论除人工神经网络之外的其他机器学习算法用于未来僵尸网络恶意软件攻击检测的性能。此外，该研究仅将性能报告为真阳性率和假阴性率。在本文中，我们建议模型的选择应该是真阳性率、假阴性率、训练准确率、测试准确率、误分类成本、预测速度和训练时间的函数。本文介绍了为风筝选择最佳机器学习模型的全面调查。在本次调查中，选择了大量机器学习算法。我们的研究表明，简单树、中树、粗树、随机欠采样提升和袋装树等树算法的变体报告了类似的有效性，但效率略有变化。最终，粗树赢得了未来僵尸网络恶意软件攻击检测的竞争和最适合算法。

关键词：网络安全、恶意软件、僵尸网络攻击、风筝、网络入侵检测。

Received: March 6, 2021 / Revised: April 4, 2021 / Accepted: May 7, 2021 / Published: June 28, 2021

About the authors: Abdullah Alabdulatif, Computer Department, College of Science and Arts in Ar Rass, Qassim University, Ar Rass, Saudi Arabia; Syed Sajjad Hussain Rizvi, Shaheed Zulfikar Ali Bhutto Institute of Science and Technology, Karachi, Pakistan; Manzoor Ahmed Hashmani, Universiti Teknologi PETRONS, Seri Iskandar, Malaysia

Corresponding author Abdullah Alabdulatif, a.alabdulatif@qu.edu.sa

1. Introduction

In the past decade, intensive growth in information and communication technology (ICT) has been observed. It includes, but is not limited to, efficient communication media, high-performance computing, massive storage units, etc. This growth has played a catalytic role in various real-world applications such as banking, finance, e-commerce, m-commerce, e-government, education, and production & service industries [1]. The scenario looks beneficial; however, due to the intensive involvement of the economic factor, it demands an efficient and robust security measure to protect data and systems [2]. Classical programmed security methods are found to be deficient in efficiency and effectiveness at the same time. They lack to handle uncertainties in real-time environments [3].

In recent literature, Machine Learning methods have bridged this gap notably. However, compared to image processing and natural language processing, the performance of machine learning methods needs significant improvement [4]. In cybersecurity, there is always a human brain against machine learning who tries to find the weakness in the machine learning methods to bypass it [5]. It has created a pressing need to devise the most efficient and robust machine learning-based cybersecurity methods to combat bot attacks or against a human attacker [6].

According to the recent literature, the common cybersecurity tasks and machine learning opportunities have three dimensions i.e. Why, What, and How. Foremost, 'Why' cover the rationale of machine learning in cybersecurity task. Specifically, it includes prediction, prevention, detection, response, and monitoring. Second, 'What's a technical layer that defines at which level to monitor issues such as a network (network traffic analysis and intrusion detection); endpoint (anti-malware); application (WAF or database firewalls); user (UBA); and process (anti-fraud). Finally, the third dimension is checking and ensuring the security of a particular area [6], [7], [8].

Kitsune, a plug-and-play NIDS was introduced in 2018 [9] as a promising light-weighted NIDS for real-time detection of online Mirai Botnet Malware attack. Kitsune is primarily based on the KitNET algorithm, which is equipped with the ensemble of artificial neural networks called 'autoencoder' to classify legitimate and suspicious network traffic. Moreover, the Kitsune Network Attack dataset was donated to the UCI machine learning repository in October 2019. This work was richly cited in the literature as a benchmark for NIDS for a real-time system. In this study, the author has compared the performance of Kitsune with Suricata, Iso. Forest, GMM, GMM Inc, and PC steam. The Kitsune has significantly outperformed as compared to these NIDS. In extension to this study, a comprehensive parametric investigation for the rationale of using an artificial neural network was observed.

Moreover, the rich dimensions of performance parameters were also a dire need. This study presents a comprehensive investigation for selecting optimal machine learning model(s) for Kitsune. In this investigation, a large set of machine learning algorithms have opted as candidate machine learning models. The selection of the model is a function of true positive rate (TPR), false-negative rate (FNR), training accuracy, test accuracy, misclassification cost, prediction speed, and train time. Our study reveals that the variants of tree algorithms such as Simple Tree, Medium Tree, Coarse Tree, RUSBoosted, and Bagged Tree have reported similar effectiveness but with higher efficiency.

2. Related Work

In recent decades, the domain of artificial intelligence and specifically machine learning and deep learning, has gained tremendous attention from researchers and developers [10], [11], [12], [13]. Specifically, cyber-security has adopted machine learning as the most exciting catalyst [14]. Likewise, the performance of network intrusion detection has significantly improved due to the foundation support of machine learning models [15], [16], [17]. The scenario looks to benefit at large. However, it is constrained specifically for the domain of computer communication networks [18]. The inter-networking devices have limited resources like storage, processing, I/O connection to handle the complex machine learning models [19]. The issue of constrained resources at interconnecting devices is further dealt with with the modern and advanced embedded system, IoT modules, and single-board computers [20-21]. In this connection, the domain of ubiquitous computing was evolved [22].

The botnet is one of the most frequent attacks reported by NIDS [23]. The researchers are investigating to find the optimum light-weighted classifier for botnet malware detection. In a study by Feizollah et al., five machine learning classifiers, namely Naïve Bayes, k-nearest neighbor, decision tree, multi-layer perceptron, and support vector machine, were evaluated for Android Malware Genome detection. This study has reported a TPR of 99.94% and an FPR of 0.06% for the kNN classifier. They concluded that the KNN is a good candidate for their dataset and application [24]. Koroniotis et al. also have investigated the play of machine learning algorithms to devise the network forensic mechanism. This mechanism is primarily based on network flow identifiers that can monitor the network's suspicious movement either by botnet or humans. This study was evaluated on the UNSW-NB15 dataset. This study also advocates for the use of machine learning algorithms [25]. In another research paper, the author has proposed a novel framework named Classification of Network

Information Flow Analysis (CONIFA). This study, too, has been evaluated on the vest set of machine learning algorithms and has concluded that machine learning algorithms for botnet malware detection could detect C&C communication channels and malicious traffic with limited devise resources [26], [27].

McKay et al. [28], Utilized the Waikato Environment for Knowledge Analysis (WEKA) data mining and analysis tool to investigate the response of various machine learning algorithms on the CICIDS2017 dataset. They have concluded that the instance-based nearest neighbor and decision tree classifiers, J48, an expanded ID3 decision tree classifier, have outperformed for real-time malware detection. The SMARTbot [29], a novel dynamic analysis framework augmented with machine learning methods to detect botnet binaries from malicious corpus, was introduced in the literature in 2016. This framework has evaluated the popular variant of artificial neural networks with back-propagation learning and variants of logistic regressions to detect malicious activities over the network. This study revealed that regression outperforms other variants of machine learning classifier for botnet apps' detection.

Moreover, they have reported an average accuracy of 99.49%. Dollah et al. also have investigated the best candidate of a machine learning algorithm for HTTP Botnet detection. This study evaluated Decision Tree, KNN, Naïve Bayes, and Random Forest classifier for HTTP Botnet detection. This study establishes that the KNN classifier has achieved an average accuracy of 92.93% with a TPR of 95.47% [30]. In another study, reported in 2018, have proposed a structural analysis-based learning framework. This framework is based on machine learning models to classify botnets and benign applications. In this study, the authors have employed Naïve Bayes, support vector machine, and REPTree to detect and classify botnets and benign applications. The authors have concluded that SVM is the best candidate for this application [31].

2.1. Research Gap and Open Area

In the light of the above literature, it can be inferred that the NIDS for botnet malware attacks essentially demands rigorous parametric evaluation on a different set of machine learning algorithms. This evaluation essentially results in selecting the best candidate machine learning algorithm for the specific NIDS and dataset. However, the rigorous parametric evaluation of Kitsune was not very well established in the respective publication. The principal contribution of this work is the comprehensive investigation for the selection of optimal machine learning model(s) for Kitsune. In this investigation, a large set of machine learning algorithms have opted. The selection of the model is a function of true positive rate (TPR), false-negative rate (FNR), training accuracy, test accuracy, misclassification cost, prediction speed, and train time. Our study reveals that the variants of tree algorithms such as Simple Tree, Medium Tree, Coarse Tree, RUSBoosted, and Bagged Tree have reported similar effectiveness but with higher efficiency.

3. Dataset Description and System Setup

The dataset of Kitsune Network Attack dataset was donated to the UCI machine learning repository and was publically available for evaluation in 2019. So far, many researchers have opted and cited the said dataset for their investigation on NIDS. It makes this dataset reportedly a benchmark dataset for NIDS. This dataset primarily collects four attack types, namely, Recon., Man in the Middle, Denial of Service, and Botnet Malware. In this study, the Botnet Malware dataset is taking into considerations. This dataset has 7.64K instances and 118 input attributes. The dataset is randomly divided into 70% training samples and 30% testing samples. The experimentation was performed on a high-performance computing machine with Core i7-7700 CPU (8 CPU) ~ 3.6 GHz, 32 GB RAM, Windows 10 Pro 64-bit, and a high-performance graphics card. Table 1 illustrates the different variants of network attacks and types present in the Kitsune dataset. Specifically, the Mirai Botnet attack is under consideration in this study.

Table 1 Kitsune datasets description

Attack Type	Attack Name	Description
Botnet Malware	Mirai	It is the set of instances and attributes that infects IoT with the Mirai malware by exploiting default credentials and then scans for new vulnerable victims network
Recon	OS Scan	A real-time Scans of the network and host operating systems to find the potential vulnerabilities
	Fuzzing	To search for the potential vulnerabilities in the camera's web servers by initiating the random commands
Man in the Middle	Video Injection	It contains the set of injected recorded video clip into a session of live streaming

	ARP MitM	This dataset contains all LAN traffic via an ARP poisoning attack
	Active Wiretap	This dataset contains all LAN traffic via active wiretap
Denial of Service	SSDP Flood	The set of instances that overloads the DVR by causing cameras to spam the server with UPnP advertisements
	SYN DoS	The set of instances that disables a camera's video stream by overloading its web server
	SSL Renegotiation	The set of instances that disables a camera's video stream by sending many SSL renegotiation packets to the camera

4. Simulation Results and Analysis

This section of the manuscript illustrates the comprehensive parametric evaluation of 15 machine learning algorithms as a function of TPR, FNR, Training Accuracy, Test Accuracy, Mis-classification cost, prediction speed, and Training Time. Moreover, the confusion matrix of each respective algorithm is also mentioned in Table 2. The dataset was divided into 70% training data and 30% testing data. The Test accuracy was computed on distinct test data, while the rest of the parameters are computed on the training data. The pictorial competition of the given machine learning algorithms against each parameter is also depicted in Fig. 1 and Fig. 2. Moreover, the testing curve is illustrated in Fig. 3. Finally, the accumulated comparison is established in Fig. 4 for ready reference.

Table 2 Parametric performance comparison of machine learning algorithm

Algorithm	*Confusion Matrix (%)*		*Predicted Class* 1	2	*True Positive (TPR) (%)*	*False Negative Rate (FNR) (%)*	*Accuracy of No-Attack*	*Accuracy of Attack*	*Net Accuracy (%)*	*Test Accuracy (%)*	*Misclassification Cost*	*Prediction Speed (Obs/sec)*	*Train Time (sec)*
Fine Tree	*True Class*	1	100	0	100	0	100	100	100	100	0	940000	791
		2	0	100	100	0							
Medium Tree	*True Class*	1	100	0	100	0	100	100	100	100	0	890000	756
		2	0	100	100	0							
Coarse Tree	*True Class*	1	100	0	100	0	100	100	100	100	0	1000000	618
		2	0	100	100	0							
Linear SVM	*True Class*	1	100	0	100	0	100	99.5	99.6	99.57	1953	1100	5767
		2	0.5	99.5	99.5	0.6							
Quadratic SVM	*True Class*	1	100	0	100	0	100	99.7	99.8	99.71	1235	2400	7196
		2	0.3	99.7	99.7	0.3							
Cubic SVM	*True Class*	1	0.7	99.3	0.7	99.3	0.7	100	84.2	84.21	79155	200000	11968
		2	0	100	100	0							
Fine Gaussian SVM	*True Class*	1	99.9	0.1	99.9	0.1	99.9	99.6	99.7	99.1	1610	350	18451
		2	0.4	99.6	99.6	0.4							
Medium Gaussian SVM	*True Class*	1	99.9	0.1	99.9	0.1	99.9	99.4	99.5	99.34	2568	470	13676
		2	0.6	99.4	99.4	0.6							
Coarse Gaussian SVM	*True Class*	1	99.9	0.1	99.9	0.1	99.9	98.9	99	99.98	4808	460	17850
		2	1.1	98.9	98.9	1.1							
Boosted Tree	*True Class*	1	0	100	0	100	0	100	84.1	84.1	79695	1200000	198
		2	0	100	100	0							
Bagged Tree	*True Class*	1	100		100	0	100	100	100	100	0	120000	273
		2		100	100	0							
Subspace Discriminate	*True Class*	1	1	99.9	0.1	99.9	1	100	84.1	84.2	79640	9300	358
		2	0	100	100	0							
RUSBoosted Tree	*True Class*	1	100	0	100	0	100	100	100	99.8	2	580000	260
		2	0	100	100	0							
Logistic Regression	*True Class*	1	0.1	99.9	0.1	99.9	0.1	100	84.1	84.13	NA	360000	1524
		2	0	100	100	0							
Gaussian Naive Bayes	*True Class*	1	100	0	100	0	100	93	94.1	94.1	29304	220000	836
		2	7	93	93	7							

5. Empirical Comparison of Performance Parameters

In Table 1, the Fine Tree Algorithm (FT) has reported 100% TPR and 0% FNR. It turns into the average accuracy of 100% with '0' misclassification cost. It also gives 940000 obs/sec the prediction speed at the minimal training cost of 791 sec. The comparable accuracies have been observed in the other variant of the Tree algorithm, i.e., Medium Tree (MT) and Coarse Tree (CT). However, the CT has reported a relatively very high prediction speed of 1000000 Obs/sec at a relatively low cost of 618 sec. Therefore among FT, MT, and CT, the CT the CT has won the competition between FT, MT, and CT due to high prediction speed, i.e., 06%, 11% high prediction speed compared to FT, and MT, respectively. The Bagged Tree, RUSBoosted Tree also has reported identical accuracy to FT, MT, and CT and low training time. Their prediction has, on average, 60% to 65% reduction in the prediction speed.

The variants of SVM, such as Linear SVM, Quadratic SVM, Fine Gaussian SVM, Medium Gaussian SVM, and Coarse Gaussian SVM, have reported the accuracy and confusion matrix relatively close to FT, MT, and CT. However, they have presented a significantly low prediction speed (about 99% decline), very high misclassification cost, and training time compared to FT, MT, and CT. The cubic SVM, Boosted Tree, Subspace Discriminant, Logistic Regression, and Gaussian Naïve Bayes have good prediction speed, but their very low TRP make them out of competition. With this analysis, the CT is turn out to the best algorithm for Mirai botnet malware detection. The above finding can be summarized in Table 3, where performances are modeled as a subjective measure.

Table 3 Subjective evaluation

ML Algorithm	*Accuracy (Training & Testing)*	*Prediction Speed*	*Misclassification cost*	*Training Time*
FT	Excellent	Good	Excellent	Fair
MT	Excellent	Fair	Excellent	Good
CT	Excellent	Excellent	Excellent	Excellent
Bagged Tree	Excellent	Poor	Excellent	Excellent
RUSBoosted	Excellent	Poor	Excellent	Excellent
Linear SVM	Good	Not Acceptable	Poor	Poor
Quadratic SVM	Good	Not Acceptable	Poor	Poor
Fine Gaussian SVM	Good	Not Acceptable	Poor	Poor
Medium Gaussian SVM	Good	Not Acceptable	Poor	Poor
Coarse Gaussian SVM	Good	Not Acceptable	Poor	Poor
Cubic SVM	Not Acceptable	Good	Not Acceptable	Not Acceptable
Boosted Tree	Not Acceptable	Good	Not Acceptable	Good
Subspace Discriminant	Not Acceptable	Good	Not Acceptable	Good
Logistic Regression	Not Acceptable	Good	Not Acceptable	Good
Gaussian Naïve Bayes	Not Acceptable	Good	Not Acceptable	Good

6. Graphical Comparison of Performance Parameters

About Fig. 1 to Fig. 7, the following set of the algorithm has been devised based on their performance. The pictorial illustration of the comprehensive view of competition for machine learning algorithm for Mirai botnet malware attack detection is also shown in Fig. 5.

6.1. S1 (The Class Level Accuracy and Net Accuracy)=

{Fine Tree, Medium Tree, Coarse Tree, Linear SVM, Quadratic SVM, Fine Gaussian SVM, Medium Gaussian SVM, Coarse Gaussian SVM, Bagged Tree, Subspace Discriminant, RUSBoosted Tree, Gaussian Naïve Bayes

6.2. S2 (Test Accuracy)=

{Fine Tree, Medium Tree, Coarse Tree, Linear SVM, Quadratic SVM, Fine Gaussian SVM, Medium Gaussian SVM, Coarse Gaussian SVM, Boosted Tree, Bagged Tree, Subspace Discriminant, RUSBoosted Tree, Logistic Regression, Gaussian Naïve Bayes}

6.3. S3 (Misclassification Cost)=

{Fine Tree, Medium Tree, Coarse Tree, Bagged Tree, RUSBoosted Tree}

6.4. S4 (Prediction Speed)=

{Fine Tree, Medium Tree, Coarse Tree, Cubic SVM, Boosted Tree, Bagged Tree, RUSBoosted Tree, Gaussian Naïve Bayes}

6.5. S5 (Training Time)=

{Fine Tree, Medium Tree, Coarse Tree, Boosted Tree, Bagged Tree, Subspace Discriminant, RUSBoosted Tree, Gaussian Naïve Bayes}

Fig. 8 and Fig. 9 illustrate the testing curve. The x-axis represents the test instances, and the y-axis shows the actual output class. The blue curve shows the actual output in this curve, and the red curve shows the predicted output. Fig. 10 depicts the test curve of FT, MT, CT, RUS Boosted Tree, Bagged Tree, which are in the set of the suggested algorithm. It is evident from the curve that this algorithm has the best match of the actual curve and predicted curve. In Fig. 4, the test curve of Subspace Discriminant and RUSBoosted Tree shows that these algorithms have good accuracy for one class only. The same response can be observed in the training phase of the respective algorithm. Likewise, given that Medium Gaussian SVM, Linear SVM has good test accuracy but at a very high computational and misclassification cost, as shown in the training phase.

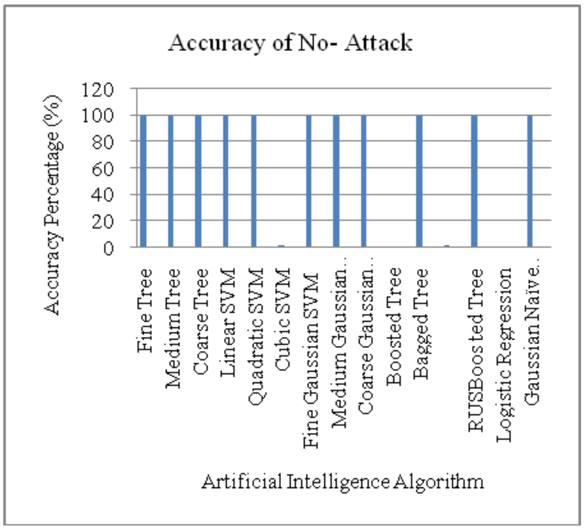


Fig. 1 Accuracy of no attack detection

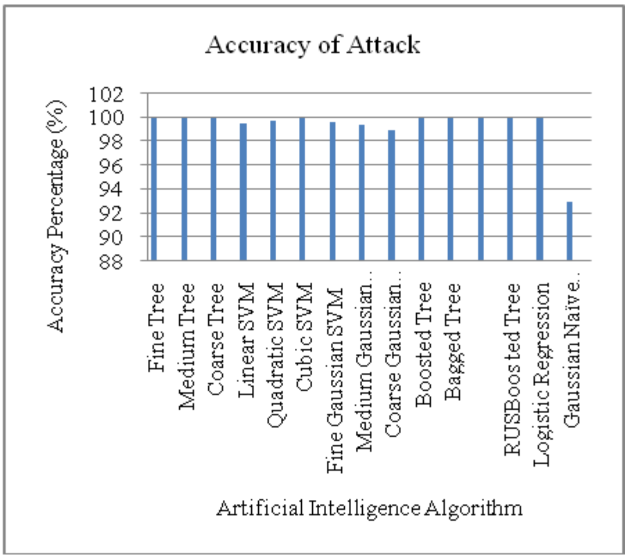


Fig. 2 Accuracy of attack detection

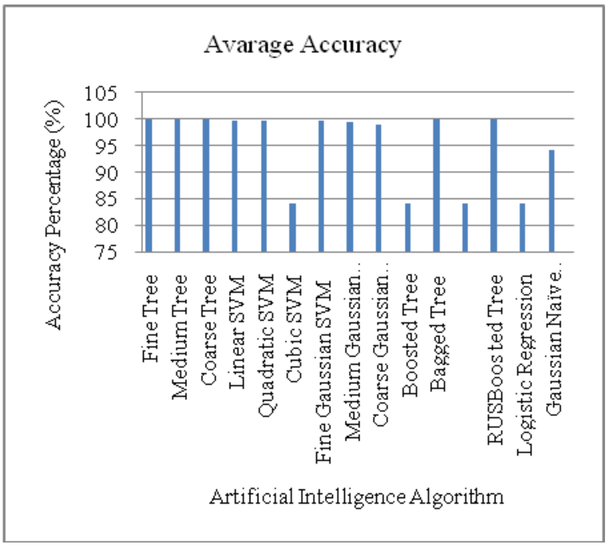


Fig. 3 Average accuracy

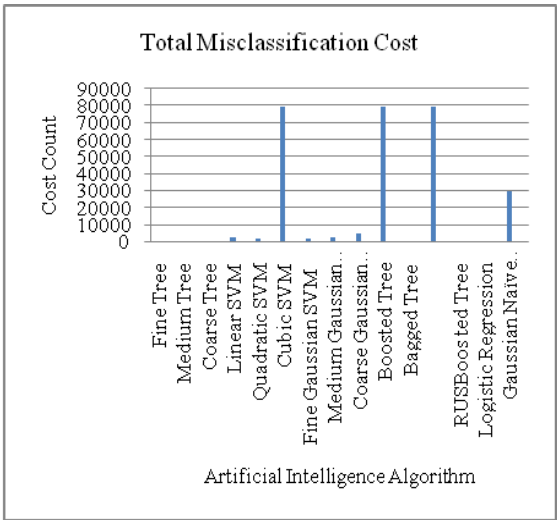


Fig. 4 Total misclassification cost

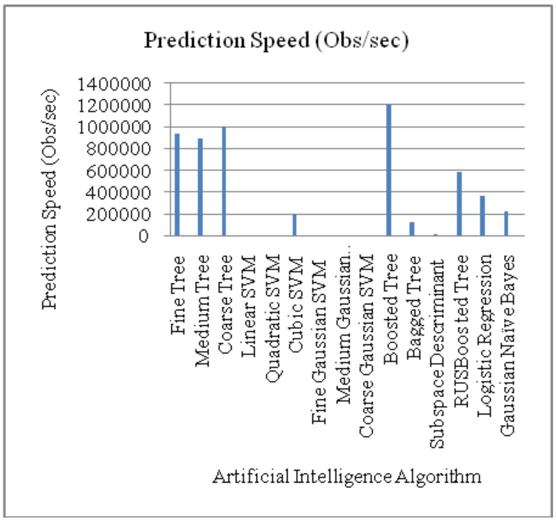


Fig. 5 Prediction speed

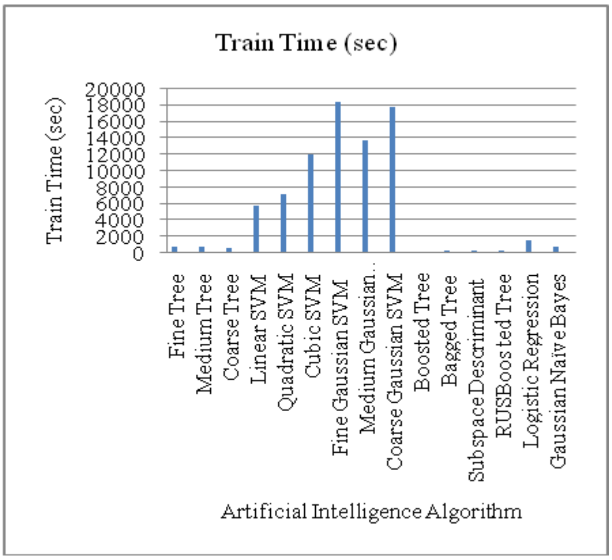


Fig. 6 Training time

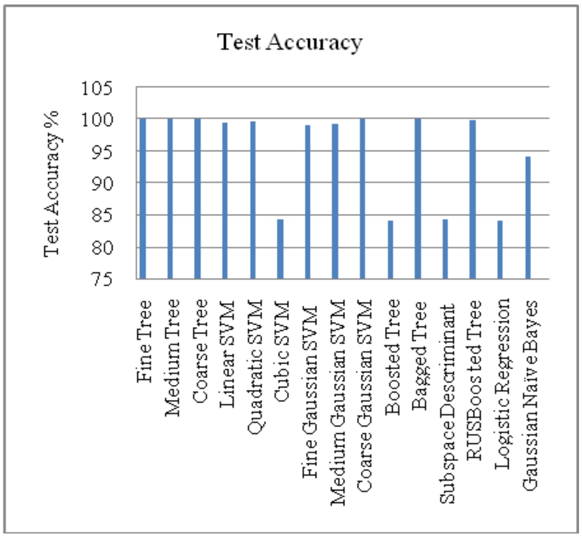
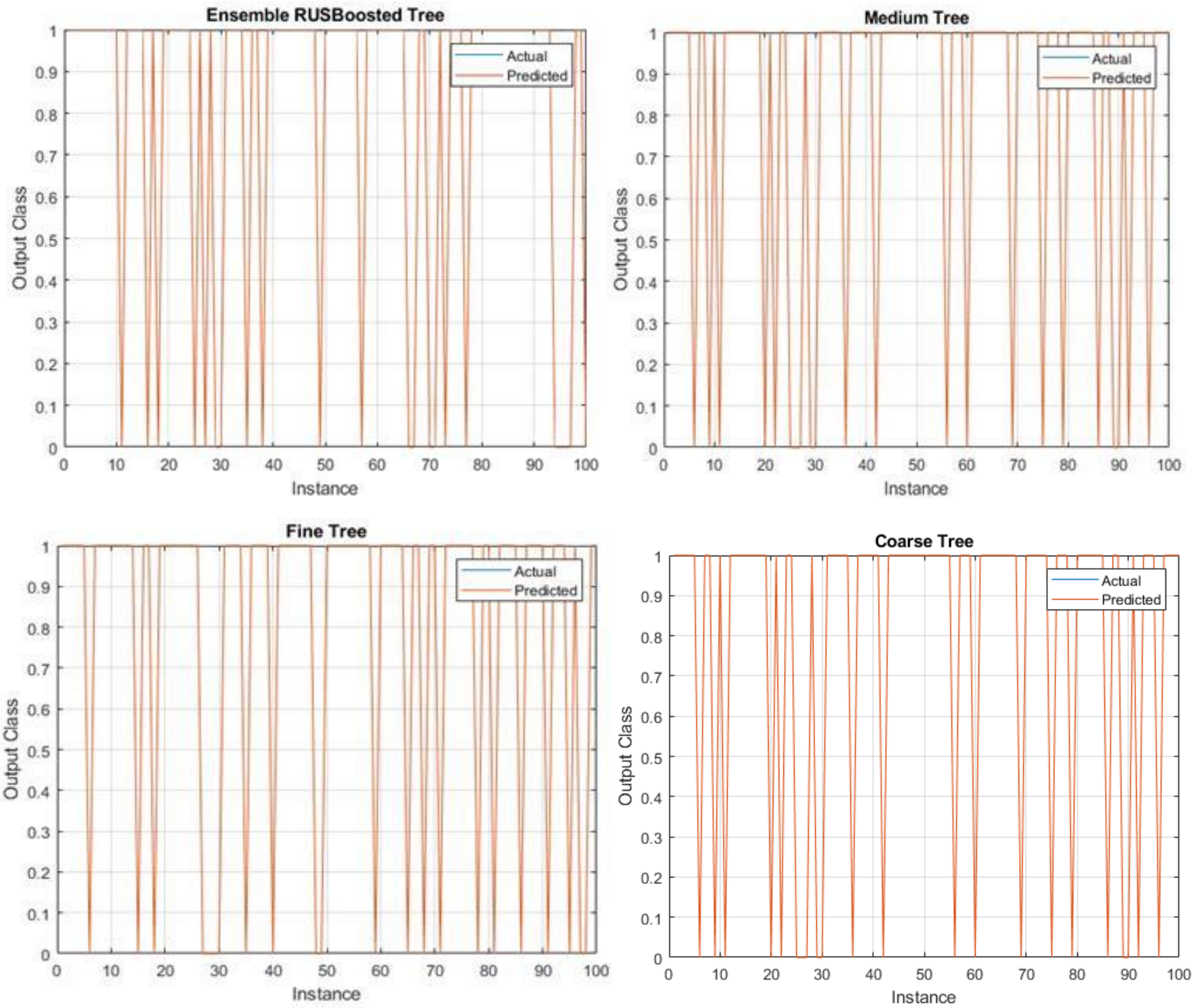


Fig. 7 Test accuracy



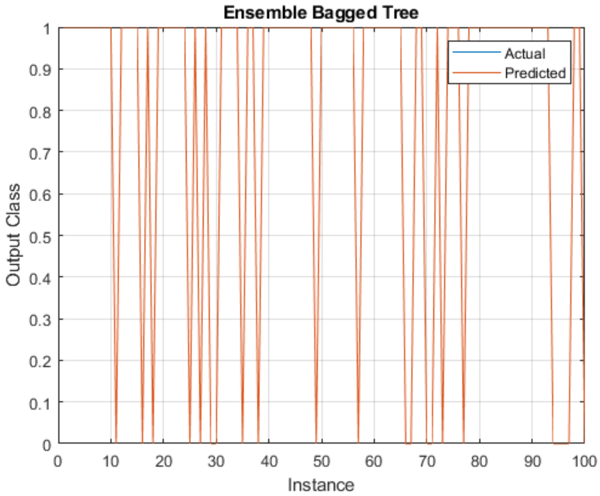


Fig. 8 Testing curve of FT, MT, CT, RUS Boosted Tree, Bagged Tree (Good alternative of ML for Miria Botnet Attack detection)

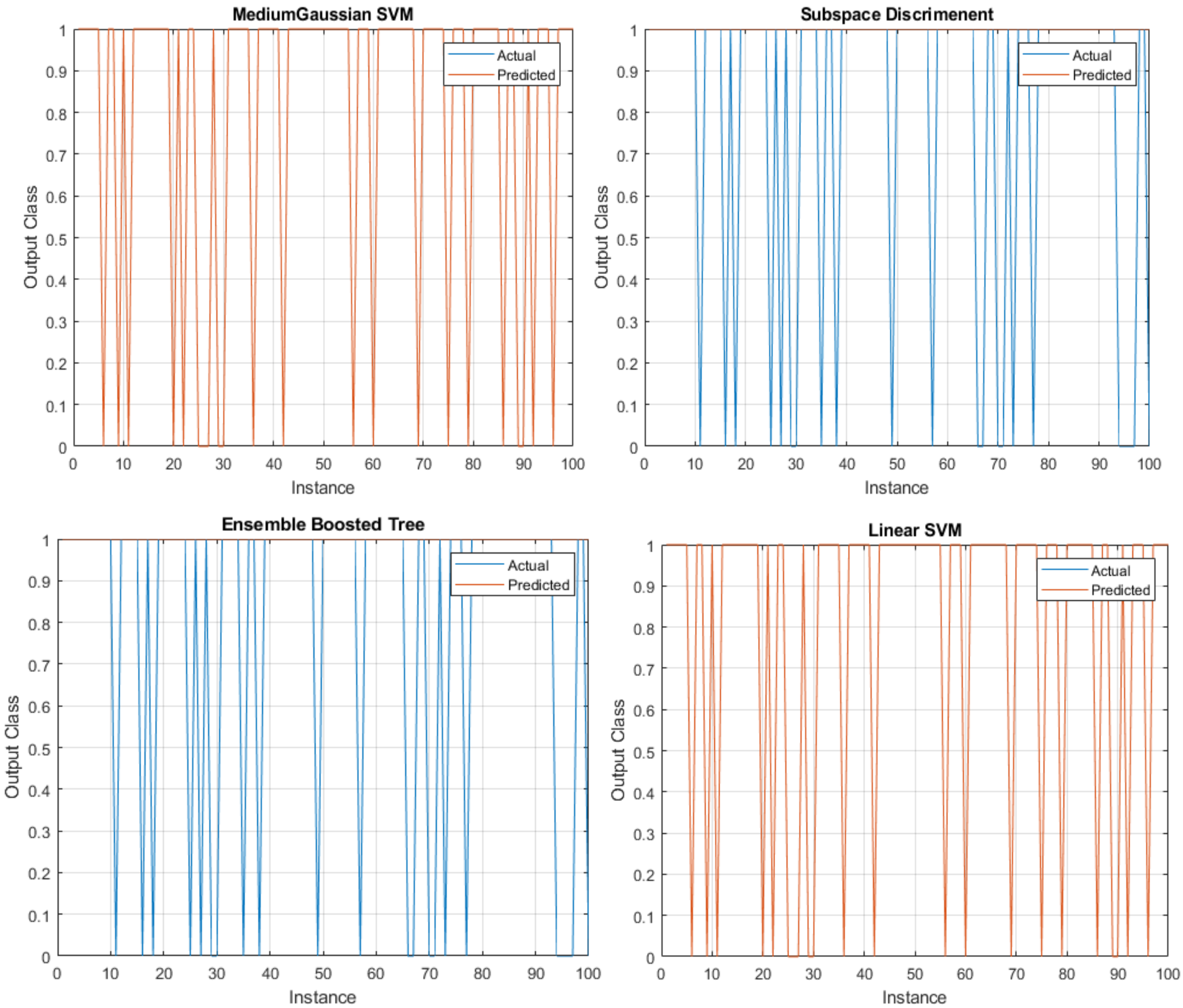


Fig. 9 Testing curve of not suggested machine learning algorithms for Miria botnet malware attack detection

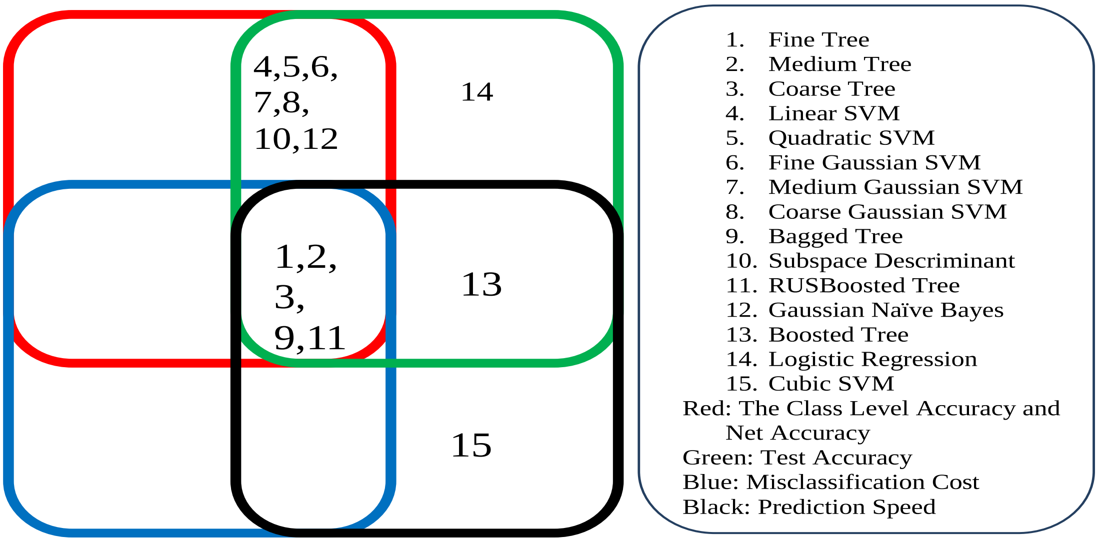


Fig. 10 The comprehensive view of competition for machine learning algorithm for Mirai botnet malware attack detection

Fig. 10 illustrates the comprehensive view of competition for machine learning algorithms for Mirai botnet malware attack detection. This figure has four quadrants of performance parameters, namely, the Class Level Accuracy and Net Accuracy (Red Color), test accuracy (Green Color), Misclassification Cost (blue color), Prediction Speed (black color). Each quadrant shows the algorithm index number best suited with the respective performance parameter. The intersection of all quadrants shows the set of the best-suited algorithm. It is inferred from this figure that algorithm index 4,5,6,7,8, 10,12 posses excellent classes level training and testing accuracy. Likewise, algorithm 1,2,3, 9,11 results with the set of optimum algorithms having excellent accuracies with almost negligible misclassification cost and efficiency as a function of prediction speed and training time. Finally, CT is ranked as the optimum algorithm for the Mirai botnet malware attack.

The simulation results are strongly advocated for the Coarse Tree for botnet malware detection. However, the scope of this work is found to be constrained due to the following reasons:

1. Due to the massive data volume, a high-performance computing machine is essential for the offline training
2. A robust and high performance embedded system(where the trained model will be deployed) would be essential for real-time testing
3. The learning scope of the application will be limited to the dimension of the given dataset

7. Conclusion

Kitsune is a plug-and-play NIDS using KitNET, based on the ensemble of artificial neural networks called 'autoencoder' to classify legitimate and suspicious network traffic. The Kitsune is a rich cited NIDS in recent literature, but its comprehensive investigation of the other machine learning algorithms was missing from the literature. Moreover, it is evident from the literature that the NIDS needs to be evaluated on the set of machine learning algorithms for the best candidate. This paper presents a comprehensive investigation for selecting optimal machine learning model(s) for Kitsune. In this investigation, a large set of machine learning algorithms have opted. The selection of the model is a function of true positive rate (TPR), false-negative rate (FNR), training accuracy, test accuracy, misclassification cost, prediction speed, and train time. Our study reveals that the variants of tree algorithms such as Simple Tree, Medium Tree, Coarse Tree, RUSBoosted, and Bagged Tree have reported similar effectiveness but with slight variation inefficiency. Finally, Coarse Tree has won the competition and best-suited algorithm for Mirai botnet malware attack detection.

References

[1] HUSSAIN S. S., HASHMANI M., MOINUDDIN M., and RAZA K. A Novel Topology in Modular ANN Approach for Multi-Modal Concept Identification and Image Retrieval. *Intelligent Automation & Soft Computing*, 2014, 20(1): 131-141. https://doi.org/10.1080/10798587.2013.863041

[2] AAMIR M., RIZVI S. S. H., HASHMANI M. A., ZUBAIR M., and AHMED J. Machine Learning Classification of Port Scanning and DDoS Attacks: A Comparative Analysis. *Mehran University Research Journal of Engineering and Technology*, 2021, 40(1): 215-229. https://doi.org/10.22581/muet1982.2101.19

[3] SEWAK M., SANJAY K. S, and HEMANT R. Comparison of Deep Learning and the Classical Machine Learning Algorithm for Malware Detection. *2018 19th International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing*. Institute of Electrical and Electronics Engineers,

Piscataway, USA, 2018. https://doi.org/10.1109/SNPD.2018.8441123
[4] LIU H., & BO L. Machine Learning and Deep Learning Methods for Intrusion Detection Systems: A Survey. *Applied Sciences*, 2019, 9(2): 1-28. https://doi.org/10.3390/app9204396
[5] LIU Q., LI P., ZHAO W., CAI, W. YU S., and LEUNG V. C. M. A Survey on Security Threats and Defensive Techniques of Machine Learning: A Data-Driven View. *Institute of Electrical and Electronics Engineers Access*, 2018, 6: 12103-12117. https://doi.org/10.1109/ACCESS.2018.2805680
[6] DUA S., & XIAN D. *Data mining and machine learning in cybersecurity*. CRC Press, Boca Raton, USA, 2016.
[7] XIN Y., KONG L., LIU Z., CHEN Y., LI Y., ZHU H., GAO M.; HOU H., and WANG C. Machine learning and deep learning methods for cybersecurity. *Institute of Electrical and Electronics Engineers Access*, 2018, 6: 35365-35381. https://doi.org/10.1109/ACCESS.2018.2836950
[8] FRALEY J. B., & CANNADY J. The Promise of Machine Learning in Cybersecurity. *SoutheastCon 2017*. Institute of Electrical and Electronics Engineers, Piscataway, USA, 2017. https://doi.org/10.1109/SECON.2017.7925283
[9] MIRSKY Y., TOMER D., YUVAL E., and ASAF S. Kitsune: an Ensemble of Autoencoders for Online Network Intrusion Detection. *Cornell University arXiv*, 2018, 1802.09089. http://dx.doi.org/10.14722/ndss.2018.23204
[10] HASHMANI M. A., JAMEEL S. M., RIZVI S. S. H., and SHUKLA S. An Adaptive Federated Machine Learning-Based Intelligent System for Skin Disease Detection: A Step Toward an Intelligent Dermoscopy Device. *Applied Sciences*, 2021, 11(5): 1-19. https://doi.org/10.3390/app11052145
[11] ALI S. E. A., RIZVI S. S. H., LAI F.-W., ALI R. F., and JAN A. A. Predicting Delinquency on Mortgage Loans: An Exhaustive Parametric Comparison of Machine Learning Techniques. *International Journal of Industrial Engineering and Management*, 2021, 12(1): 1-13. http://doi.org/10.24867/IJIEM-2021-1-272
[12] ZAFFAR M. HASHMANI M. A., SAVITA K. S., RIZVI S. S. H., and REHMAN M. Role of FCBF Feature Selection Algorithm in Educational Data Mining. *Mehran University Research Journal of Engineering and Technology*, 2020, 39(4): 772-778. https://doi.org/10.22581/muet1982.2004.09
[13] UDDIN V., RIZVI S. S. H., HASHMANI M. A., JAMEEL S. M., and ANSARI T. A Study of Deterioration in Classification Models in Real-Time Big Data Environment. *International Conference of Reliable Information and Communication Technology*. Springer, Cham, Switzerland, 2019: 79-87. https://doi.org/10.1007/978-3-030-33582-3_8
[14] APRUZZESE G., COLAJANNI M., FERRETTI L., GUIDO A., and MARCHETTI M. On the Effectiveness of Machine and Deep Learning for Cybersecurity. *2018 10th International Conference on Cyber Conflict.* Institute of Electrical and Electronics Engineers, Piscataway, USA, 2018: 371-390. https://doi.org/10.23919/CYCON.2018.8405026
[15] KUN Z., & ZHANG Q. Application of Machine Learning in Network Intrusion Detection. *Journal of Data Acquisition and Processing*, 2017, 32(3): 479-488.
[16] BISWAS S. K. Intrusion Detection Using Machine Learning: A Comparison Study. *International Journal of Pure and Applied Mathematics*, 2018, 118(19): 101-114. https://doi.org/10.1186/s40537-018-0145-4
[17] MAHFOUZ A. M., DEEPAK V., and SAJJAN G. S. Comparative Analysis of ML Classifiers for Network Intrusion Detection. *Fourth International Congress on Information and Communication Technology*. Springer, Singapore, 2020. https://doi.org/10.1007/978-981-32-9343-4_16
[18] VAN N. T., TRAN N. T., and LE T. S. An Anomaly-Based Network Intrusion Detection System Using Deep Learning. *2017 International Conference on System Science and Engineering*. Institute of Electrical and Electronics Engineers, Piscataway, USA, 2017. https://doi.org/10.1109/ICSSE.2017.8030867
[19] KATO N., MAO B., TANG F., KAWAMOTO Y., and LIU J. Ten Challenges in Advancing Machine Learning Technologies Toward 6G. *Institute of Electrical and Electronics Engineers Wireless Communications*, 2020, 27(3): 96-103. https://doi.org/10.1109/MWC.001.1900476
[20] LEE J., STANLET M., SPANIAS A., and TEPEDELENLIOGLU C. Integrating machine learning in embedded sensor systems for Internet-of-Things applications. *2016 Institute of Electrical and Electronics Engineers International Symposium on Signal Processing and Information Technology*. Institute of Electrical and Electronics Engineers, Piscataway, USA, 2016. https://doi.org/10.1109/ISSPIT.2016.7886051
[21] NADESKI M. *Bringing Machine Learning to Embedded Systems*. Texas Instruments, Dallas, USA, 2019. https://www.ti.com/lit/wp/sway020a/sway020a.pdf?ts=1623255278008&ref_url=https%253A%252F%252Fwww.google.com%252F
[22] CÁRDENAS-ROBLEDO L. A., & ALEJANDRO P.-A. Ubiquitous Learning: A Systematic Review. *Telematics and Informatics*, 2018, 35(5): 1097-1132. https://doi.org/10.1016/j.tele.2018.01.009
[23] ALIEYAN K., ALMOMANI A., MANASRAH A., and KADHUM M. M. A Survey of Botnet Detection Based on DNS. *Neural Computing and Applications*, 2017, 28(7): 1541-1558. https://doi.org/10.1007/s00521-015-2128-0
[24] DA COSTA V. G. T., BARBON S., MIANI R. S., RODRIGUES J. J. P. C., and ZARPELAO B. B. Detecting Mobile Botnets through Machine Learning and System Calls Analysis. *2017 Institute of Electrical and Electronics Engineers International Conference on Communications*. Institute of Electrical and Electronics Engineers, Piscataway, USA, 2017. https://doi.org/10.1109/ICC.2017.7997390
[25] KORONIOTIS N., MOUSTAFA N., SITNIKOVA E., and SLAY J. Towards Developing Network Forensic Mechanism for Botnet Activities in the IoT Based on Machine Learning Techniques. *International Conference on Mobile Networks and Management*. Springer, Cham, Switzerland, 2017. https://doi.org/10.1007/978-3-319-90775-8_3
[26] ABRAHAM B., MANDYA A., BAPAT R., ALALI F., BROWN D. E., and VEERARAGHAVAN M. A Comparison of Machine Learning Approaches to Detect Botnet Traffic. *2018 International Joint Conference on Neural Networks.* Institute of Electrical and Electronics Engineers, Piscataway, USA, 2018. https://doi.org/10.1109/IJCNN.2018.8489096

[27] AZAB A., MAMOUN A., and MAHDI A. Machine Learning-Based Botnet Identification Traffic. International Conference on Trust, Security and Privacy in Computing and Communications. Institute of Electrical and Electronics Engineers, Piscataway, USA, 2016. https://doi.org/10.1109/TrustCom.2016.0275

[28] MCKAY R., PENDLETON B., BRITT J., and NAKHAVANIT B. Machine Learning Algorithms on Botnet Traffic: Ensemble and Simple Algorithms. *Proceedings of the 2019 3rd International Conference on Compute and Data Analysis*. Association for Computing Machinery, New York, United States, 2019. https://doi.org/10.1145/3314545.3314569

[29] KARIM A., ROSLI S., and MUHAMMAD K. K. SMARTbot: A Behavioral Analysis Framework Augmented with Machine Learning to Identify Mobile Botnet Applications. *Public Library of Science One*, 2016, 11(3): 1-35. https://doi.org/10.1371/journal.pone.0150077

[30] DOLLAH R. F. M., FAIZAL M., ARIF F., MAS'UD M.Z., and XIN L. K. Machine Learning for HTTP Botnet Detection Using Classifier Algorithms. *Journal of Telecommunication, Electronic and Computer Engineering*, 2018, 10: 27-30. https://www.semanticscholar.org/paper/Machine-Learning-for-HTTP-Botnet-Detection-Using-Dollah-Faizal/bfa426bf57513c87f0969ba5e9e457d6f50279b6

[31] KIRUBAVATHI G., & ANITHA R. Structural Analysis and Detection of Android Botnets Using Machine Learning Techniques. *International Journal of Information Security*, 2018, 17(2): 153-167. https://doi.org/10.1007/s10207-017-0363-3

参考文:

[1] HUSSAIN S. S., HASHMANI M., MOINUDDIN M., 和 RAZA K. 一种用于多模态概念识别和图像检索的模块化人工神经网络方法的新拓扑。智能自动化与软计算，2014，20(1): 131-141. https://doi.org/10.1080/10798587.2013.863041

[2] AAMIR M., RIZVI S. S. H., HASHMANI M. A., ZUBAIR M., 和 AHMED J. 端口扫描和分布式拒绝服务攻击的机器学习分类：比较分析。梅赫兰大学工程技术研究杂志，2021，40(1): 215-229. https://doi.org/10.22581/muet1982.2101.19

[3] SEWAK M., SANJAY K. S, 和 HEMANT R. 深度学习与用于恶意软件检测的经典机器学习算法的比较。 2018 年第 19 届软件工程、人工智能、网络和并行/分布式计算国际会议。美国皮斯卡塔韦电气和电子工程师协会, 2018. https://doi.org/10.1109/SNPD.2018.8441123

[4] LIU H., 和 BO L. 入侵检测系统的机器学习和深度学习方法：一项调查。应用科学, 2019, 9(2): 1-28. https://doi.org/10.3390/app9204396

[5] LIU Q., LI P., ZHAO W., CAI, W. YU S., 和 LEUNG V. C. M. 机器学习的安全威胁和防御技术调查：数据驱动的观点。电气和电子工程师协会访问权限，2018，6: 12103-12117. https://doi.org/10.1109/ACCESS.2018.2805680

[6] DUA S., 和 XIAN D. 网络安全中的数据挖掘和机器学习. CRC 出版社，美国博卡拉顿, 2016.

[7] XIN Y., KONG L., LIU Z., CHEN Y., LI Y., ZHU H., GAO M.; HOU H., 和 WANG C. 用于网络安全的机器学习和深度学习方法。电气和电子工程师协会访问权限, 2018, 6: 35365-35381. https://doi.org/10.1109/ACCESS.2018.2836950

[8] FRALEY J. B., 和 CANNADY J. 机器学习在网络安全中的前景。2017年东南会议。美国皮斯卡塔韦电气和电子工程师协会, 2017. https://doi.org/10.1109/SECON.2017.7925283

[9] MIRSKY Y., TOMER D., YUVAL E., 和 ASAF S. Kitsune：用于在线网络入侵检测的自动编码器集合。康奈尔大学档案, 2018, 1802.09089. http://dx.doi.org/10.14722/ndss.2018.23204

[10] HASHMANI M. A., JAMEEL S. M., RIZVI S. S. H., 和 SHUKLA S. 基于自适应联合机器学习的皮肤疾病检测智能系统：迈向智能皮肤镜设备的一步。应用科学, 2021, 11(5): 1-19. https://doi.org/10.3390/app11052145

[11] ALI S. E. A., RIZVI S. S. H., LAI F.-W., ALI R. F., 和 JAN A. A. 预测抵押贷款拖欠：机器学习技术的详尽参数比较。国际工业工程与管理杂志, 2021, 12(1): 1-13. http://doi.org/10.24867/IJIEM-2021-1-272

[12] ZAFFAR M. HASHMANI M. A., SAVITA K. S., RIZVI S. S. H., 和 REHMAN M. 基于快速相关的过滤器特征选择算法在教育数据挖掘中的作用。梅赫兰大学工程技术研究杂志, 2020, 39(4): 772-778. https://doi.org/10.22581/muet1982.2004.09

[13] UDDIN V., RIZVI S. S. H., HASHMANI M. A., JAMEEL S. M., 和 ANSARI T. 实时大数据环境中分类模型退化的研究。可靠的信息和通信技术国际会议。斯普林格，湛，瑞士，2019: 79-87. https://doi.org/10.1007/978-3-030-33582-3_8

[14] APRUZZESE G., COLAJANNI M., FERRETTI L., GUIDO A., 和 MARCHETTI M. 关于机器和深度学习对网络安全的有效性。2018第十届网络冲突国际会议。美国皮斯卡塔韦电气和电子工程师协会, 2018: 371-390. https://doi.org/10.23919/CYCON.2018.8405026

[15] KUN Z., 和 ZHANG Q. 机器学习在网络入侵检测中的应用。数据采集与处理杂志, 2017, 32(3): 479-488.

[16] BISWAS S. K. 使用机器学习进行入侵检测：比较研究。国际纯粹与应用数学杂志, 2018, 118(19): 101-114. https://doi.org/10.1186/s40537-018-0145-4

[17] MAHFOUZ A. M., DEEPAK V., 和 SAJJAN G. S. 用于网络入侵检测的机器学习分类器的比较分析。第四届国际信息和通信技术大会。新加坡施普林格, 2020. https://doi.org/10.1007/978-981-32-9343-4_16

[18] VAN N. T., TRAN N. T., 和 LE T. S. 使用深度学习的基于异常的网络入侵检测系统。2017年系统科学与工程国际会议。美国皮斯卡塔韦电气和电子

工程师协会, 2017. https://doi.org/10.1109/ICSSE.2017.8030867
[19] KATO N., MAO B., TANG F., KAWAMOTO Y., 和 LIU J. 推动机器学习技术迈向6代的十大挑战。电气和电子工程师协会无线通信, 2020, 27(3): 96-103. https://doi.org/10.1109/MWC.001.1900476
[20] LEE J., STANLET M., SPANIAS A., 和 TEPEDELENLIOGLU C. 将机器学习集成到用于物联网应用的嵌入式传感器系统中。2016年电气和电子工程师协会信号处理和信息技术国际研讨会。美国皮斯卡塔韦电气和电子工程师协会, 2016. https://doi.org/10.1109/ISSPIT.2016.7886051
[21] NADESKI M. 将机器学习引入嵌入式系统。德州仪器，美国达拉斯, 2019. https://www.ti.com/lit/wp/sway020a/sway020a.pdf?ts=1623255278008&ref_url=https%253A%252F%252Fwww.google.com%252F
[22] CÁRDENAS-ROBLEDO L. A., 和 ALEJANDRO P.-A. 无处不在的学习：系统回顾。远程信息处理和信息学, 2018, 35(5): 1097-1132. https://doi.org/10.1016/j.tele.2018.01.009
[23] ALIEYAN K., ALMOMANI A., MANASRAH A., 和 KADHUM M. M. 基于域名系统的僵尸网络检测综述。神经计算与应用, 2017, 28(7): 1541-1558. https://doi.org/10.1007/s00521-015-2128-0
[24] DA COSTA V. G. T., BARBON S., MIANI R. S., RODRIGUES J. J. P. C., 和 ZARPELAO B. B. 通过机器学习和系统调用分析检测移动僵尸网络。 2017年电气与电子工程师学会国际通信会议。美国皮斯卡塔韦电气和电子工程师协会, 2017. https://doi.org/10.1109/ICC.2017.7997390
[25] KORONIOTIS N., MOUSTAFA N., SITNIKOVA E., 和 SLAY J. 基于机器学习技术为物联网中的僵尸网络活动开发网络取证机制。移动网络和管理国际会议。斯普林格，湛，瑞士, 2017. https://doi.org/10.1007/978-3-319-90775-8_3
[26] ABRAHAM B., MANDYA A., BAPAT R., ALALI F., BROWN D. E., 和 VEERARAGHAVAN M. 检测僵尸网络流量的机器学习方法的比较。2018年国际神经网络联合会议。美国皮斯卡塔韦电气和电子工程师协会, 2018. https://doi.org/10.1109/IJCNN.2018.8489096
[27] AZAB A., MAMOUN A., 和 MAHDI A. 基于机器学习的僵尸网络识别流量。计算和通信中的信任、安全和隐私国际会议。美国皮斯卡塔韦电气和电子工程师协会, 2016. https://doi.org/10.1109/TrustCom.2016.0275
[28] MCKAY R., PENDLETON B., BRITT J., 和 NAKHAVANIT B. 僵尸网络流量的机器学习算法：集成算法和简单算法。2019年第三届计算与数据分析国际会议论文集。计算机协会，纽约，美国, 2019. https://doi.org/10.1145/3314545.3314569
[29] KARIM A., ROSLI S., 和 MUHAMMAD K. K. 智能机器人：通过机器学习增强行为分析框架以识别移动僵尸网络应用程序。公共科学图书馆一, 2016, 11(3): 1-35. https://doi.org/10.1371/journal.pone.0150077
[30] DOLLAH R. F. M., FAIZAL M., ARIF F., MAS'UD M.Z., 和 XIN L. K. 使用分类器算法进行超文本传输协议僵尸网络检测的机器学习。电信、电子与计算机工程杂志, 2018, 10: 27-30. https://www.semanticscholar.org/paper/Machine-Learning-for-HTTP-Botnet-Detection-Using-Dollah-Faizal/bfa426bf57513c87f0969ba5e9e457d6f50279b6
[31] KIRUBAVATHI G., 和 ANITHA R. 使用机器学习技术对安卓僵尸网络进行结构分析和检测。国际信息安全杂志, 2018, 17(2): 153-167. https://doi.org/10.1007/s10207-017-0363-3