Mesopotamian journal of Cybersecurity
Vol.2021, **pp.** 9–15
DOI: https://doi.org/10.58496/MJCS/2021/003; ISSN: 2958-6542
https://mesopotamian.press/journals/index.php/CyberSecurity

Research Article

Improved feature selection method for features reduction in intrusion detection systems

Naeemullah Khan[1], Ismael Khaleel [2,*], Elika Daghighi [3]

[1] *Thapar Institute of Engineering and Technology, India*

[2] *Modern University of Business and Science (MUBS), Lebanon*

[2] *Technical and Vocational University, Tehran, Iran*

ARTICLE INFO

Article History

Received 23 Nov. 2020
Accepted 5 Jan 2021

Keywords

Feature Selection

SVM

IDS

ABSTRACT

Many methods have been used to build intrusion detection systems based on the intended aim to be achieved in with the selected method. The hybrid methods (more than one method) usually give better results and accuracy. The recent developments and popularisation of network & information technologies have necessitated the need for network information security. Human-based smart intrusion detection systems (IDSs) are built with the capability to either warn or intercept network intrusion; this is not possible with conventional network security systems. However, most information security studies have focused on the improvement of the effectiveness of smart network IDSs. This study used the TLBO algorithm as a feature selection algorithm to choose the best subset features and the SVM classifier to classify the packet if it is an intrusion or a normal packet. Two machine learning datasets were used to test the proposed algo; the results show that the proposed algorithm performs better than many of the existing works in IDS.

1. INTRODUCTION

Cyber-attack refers to the deliberate alteration, disruption, destruction, deception, and degrading of computer networks or systems by an intruder over a sustained period [1, 2]. For the accurate detection of network attacks, several approaches have been developed with the aim of detecting, preventing, and reducing cyber-attack-related damages to computer systems/networks [3]. Such methods include the use of firewalls, intrusion detection systems (ISD) and intrusion prevention systems (IPS). Cyber-attacks can be launched in several ways; however, system and network intrusions remain the commonest forms of cyber-attack [4]. Regarding IDS, they are either hardware or software systems that can monitor and detect network intrusions in real time to prevent network exploitation [5]. The financial implication of cyber-attacks on U.S. organisations has been quantified in many studies [6]. Research also indicated that cyber-attacks were increasing in frequency and complexity, growing by 11% between 2012 and 2013 in U.K. small businesses [7] and from 1,334 incidents in 1993 to 137,529 incidents in 2003[8], resulting in a need for improved defence mechanisms [9]. Increased dependence on information across countries, organisations, and the military is a reason given for the need for improved methods of blocking cyber-attacks [10]. This dependence on information increases the overall value to the organisation . Being that organisations mostly rely on information, it has become necessary to ensure adequate security of such information for the safety of society and the economy [11]. Both network and system intrusions are greatly increasing in complexity and frequency at the same time. Before now, network attacks were targeted at just a single system component, but today, network attacks involve several techniques to target numerous segments of the network [12]. For instance, the viruses available today are more complicated when compared to those of the early years of computing; this has increased the challenges related to detecting and eliminating such viruses from computer systems [13]. A prominent example of the increased complexity of multi-layered approach cyberattacks is the 2010 Stuxnet attack at a nuclear fuel processing plant in Iran which was targeted against the industrial control systems. IDS is a commonly used method to prevent or reduce cyber-attacks; they may require time to efficiently identify an attack, starting from when the attack was launched

[*] *Corresponding author. Email: ismai.khaleel70@gmail.com*

to its recognition (this period is called the detection latency). This speed of detection, called the detection latency, is a severe challenge in IDS design [5] because if the latency is significant, it could provide attackers with ample time to cause much damage to the system even before detecting their presence. Hence, it is desired that IDS exhibit faster latency and increased sensitivity to ensure real-time attack detection. Such increased sensitivity would ensure faster cyber-attacks detection. A research gap exists regarding information showing if cyber-attack-related damage is more significant to the existing IDS latency than those with lower latency. This study aims to speed up the detection time (minimise latency).

2. BACKGROUND OF THE STUDY

Cyber-attacks present a two-part problem as they are being launched more frequently, and their design is becoming more sophisticated. To date, research into cyber-attacks has identified that cyber-attacks are increasing in frequency and, at the same time, in complexity[14]. These cyber-attacks are intentionally designed to do the most significant possible damage to the victim organisation [15]. if latency is correlated with cyber-attack damage. If this is the case, a reduction in latency could yield a reduction in attack damage. Seminal research in cyber-attacks includes research by Dorothy Denning: An Intrusion-Detection Model[16]. This was one of the first models developed for IDS systems and included extensive descriptions of attack types and terminology used in describing cyber-attacks. Intrusion Detection Systems: A Survey and Taxonomy [17] are widely cited in the scholarly literature as it contained an early taxonomy for IDS systems that were being utilised at the time and also outlined areas for future research into IDS systems. Several theories are associated with cyber-attacks and intrusion detection systems.

3. PROBLEM STATEMENT

Several types of latency can affect intrusion detection systems. Network latency is defined as the delay from the initial transmission of the packet header at the source to the reception of the end of the packet at the destination[18].This encompasses the very first part of a transmission until the very end of the transmission. It indicates how long it takes to transmit information between two locations and is a measure of that delay. In contrast, latency period, or detection latency as used in this research, is a measure of the time between when an attack starts and the system's detection of the event. A third latency factor is the period between the start of the attack and the start of corrective action by system administrators . All three of these latency factors can increase in duration as network traffic increases and the workload approaches a saturation point[19]. Detection latency is seldom used for measuring IDS systems but is important to study. A goal used in past research and IDS development was a 100% detection rate. However, a need is being recognised to reduce the amount of time that it takes to detect intruders to limit the amount of potential damage done. Detection latency is critical and should be developed as a key metric[20]. This identified need is the basis for the research in this dissertation. One technology used to handle bursts of network activity is to add the ability to store traffic data and delay the processing of it until later. These storage IDSs can delay checks until the devices are less busy or idle, with a tradeoff of more extended detection latency [21].With storage-based IDS systems, one option is to use a pull-based approach where information is downloaded at specific intervals. If these intervals are close together, detection processing increases and latency may be reduced, but the latency increases if the intervals are lengthened [22]. A limitation of this approach is that attackers have a longer time to work before being detected.

4. CLASSIFICATION PROCESS

Classification is an essential step in the proposed technique; it is executed in two phases to achieve precision, accuracy, and faster classification. Hence, the procedure is facilitated by completing further analyses only when the network is under active attack. The proposed approach combines two major classification methods (RF & PART). Figure 1 presents the central concept of this stage and how the classification process is analyzed. Breiman [11] presented RF as a cooperative learning technique that produces different classifiers & summarizes the outcomes. Additionally, it is executable via two major procedures (boosting and bagging) if needed to perform the prediction and classification tasks [10].

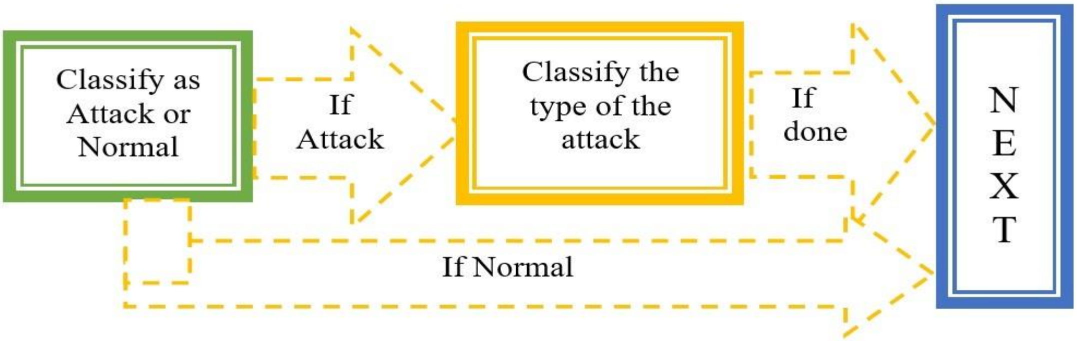

Fig.1. Classification Process

In boosting, additional weights are assigned by the succeeding trees to wrongly classified instances during initial trials; the weighted score for classification task is calculated at the end. Contrarily, in bagging, there is no relationship between the succeeding trees and the previous ones; each tree is grown using a bootstrap sample. Classification is achieved using the majority score split concept (Liaw & Wiener 2002); the RF grows numerous trees, each producing a classification and each class is assigned a specific score. Hence, the forest portrays the classification with the maximum score. The term was coined from the random DF employed by Tin (1995). Random forest (RF) combines tree predictors in a manner that each tree is dependent on the values of a random vector sampled independently; all the trees in the forest have the same distribution. The classification error of tree classifiers mainly comes from the score of each tree in the forest and the inter-dependence among them. The RF process in this study initiates with the creation of numerous trees; it introduces randomness into trees such that the correlation between the trees is minimized. The formation of each tree in the collection begins with the random selection of a small group of the input characteristics to split at each node, followed by calculating the best split based on the training set features. A two-step randomization technique will be used in the splitting process; initially, a bootstrap sample is used to grow the tree before introducing another randomization stage using a random feature selection technique. In a summary, we will randomly select a subset of m-tries at each tree node rather than splitting the tree node using all features of k, where m ∈ [1,k] for node splitting. The concept is re-sampling the training data set with replacement upon building a tree. Studies have investigated the consistency of RF in setting conditional quantile prediction [45]. The RF procedure begins with a new random vector n creation for each nth tree independent of the previous random vectors θ1,θ2.... It is generated from the same distribution; a tree is grown based on the training set θn. As a result, the tree organization of RF is dependent on the classifiers {h(X,θn,n = 1,2,3}, where {θn} represent the random vectors. A score is assigned to each tree as earlier described. For a network under attack, the partial decision tree PART [26][17] method will be employed for the classification task owing to its numerous advantages over the other techniques. The network will be labelled with the type of attack so that the process will be performed faster than RF. PART algorithm was employed in this study instead of RF because it is a rule-based method and does not need to reach global optimization to achieve accurate results; again, this will improve the speed of the classification process. PART can label new occurrences in a faster way with good level of accuracy and precision. Furthermore, PART relies on the divide-and-rule approach; hence, its covered alternatives are eliminated once a rule is constructed. It keeps repeated the rule creation process for the remaining instances all the possible outcomes are executed. This implies that creating a single rule requires building a pruned DT for the current cases set. Consequently, the leaf with the highest coverage will be transformed into a rule, and the tree will be discarded later.

The notion of persistently building DT that the majority will be rejected in PART seems not as unusual as it portrays. A rule can be obtained by employing a pruned tree instead of incrementally building it via step-wise addition of combinations. PART can avoid the issue of over-pruning of the rule learner-divide-and-rule concept [33]. The execution speed of the model is significantly improved; the basic idea is to build a partial DT rather than a fully explored DT. A partial DT is a regular DT that builds divisions to unknown sub-trees. Upon building a partial tree, a single rule is generated based on that tree [43]. The aim is to establish the ultimate general rule via selecting the leaf that covers the maximum number of instances or the leaf with the least error rate.

5. RESULTS AND DISCUSSION

The parameter settings for the simulation model are shown in Table I.

TABLE I. PARAMETER SETTINGS

Parameter	Value
Population size	20
Number of generations	40
Crossover type	Half-uniform
Mutation type	Bit-flip
Size of training data	30% (Random)
Size of testing data	100% (Full dataset)

Tables 4 and 5 present the accuracy results for both datasets, respectively. The accuracy result of the KDDCUP99 dataset is presented in Table II.

TABLE II. ACCURACY RESULT OF THE KDDCUP99 DATASET

Classifier	**TLBO**	
LR	No. of features	Accuracy
	3	0.995
Total time	12.2512	
SVM	3	0.995
	6	1.00
Total time	2382.3301	
ELM	3	0.97
	4	0.99
	5	0.995
	8	1.00
Total time	4.0717	

As seen in Table 4, For each ML, the number of features, the accuracy, and the execution time were calculated. The numbers in red are the best results for TLBO. TLBO consistently presents better accuracies than other algorithms using the three ML techniques and better time accuracy using LR and SVM ML techniques. However, TLBO exhibits a better execution time with ELM. The results of the CICIDS2017 dataset are presented in Table III.

On the CICIDS 2017 dataset, TLBO consistently shows good accuracy using the three ML techniques. matrix for the best accuracy from the application of ITLBO to the KDDCUP99 dataset (Table IV). Table V shows the confusion matrix for the CICIDS dataset.

TABLE III. ACCURACY RESULT OF THE CICIDS 2017 DATASET

Classifier	**TLBO**	
LR	No. of features	Accuracy
	14	0.94
	15	0.965
	27	0.97
Total time	33.06	
SVM	24	0.84
	26	0.92
Total time	4161.3924	
ELM	13	0.86
	15	0.885
	16	0.905
	19	0.91
	20	0.92
Total time	3.4071	

TABLE IV. KDDCUP99 CONFUSION MATRIX

	Predicted intrusion	**Predicted normal**
Actual intrusion	3,923,506	2,144
Actual normal	83	972,698

TABLE V CICIDS DATASET CONFUSION MATRIX

	Predicted normal	**Predicted intrusion**
Actual normal	2,606,223	25,402
Actual intrusion	58,719	367,159

TABLE VI. RESULTS

	KDDCUP99	CICIDS2017
Detection rate	0.9995	0.99
Error rate	0.0045	0.027
FPR	0.000085	0.13
FNR	0.00054	0.0096
TPR	0.999	0.99
Precision	0.9995	0.99
F-measure	0.998	0.99

Funding

None

Conflicts Of Interest

The authors declare no conflicts of interest.

Acknowledgement

The authors would like to thank the anonymous reviewers for their valuable comments.

References

[1] M. A. Mohammed, Z. H. Salih, N. Ţăpuş, and R. A. K. Hasan, "Security and accountability for sharing the data stored in the cloud," in *RoEduNet Conference: Networking in Education and Research, 2016 15th*, 2016, pp. 1-5: IEEE.

[2] H. Debar, M. Dacier, and A. Wespi, "A revised taxonomy for intrusion-detection systems," in *Annales des télécommunications*, 2000, vol. 55, no. 7-8, pp. 361-378: Springer.

[3] S. K. Gautam and H. Om, "Computational neural network regression model for Host-based Intrusion Detection System," *Perspectives in Science*, vol. 8, pp. 93-95, 2016.

[4] I. Manzoor and N. Kumar, "A feature reduced intrusion detection system using ANN classifier," *Expert Systems with Applications*, vol. 88, pp. 249-257, 2017.

[5] R. A. R. Ashfaq, Y.-l. He, and D.-g. Chen, "Toward an efficient fuzziness based instance selection methodology for intrusion detection system," *International Journal of Machine Learning and Cybernetics*, vol. 8, no. 6, pp. 1767-1776, 2017.

[6] Z. Chiba, N. Abghour, K. Moussaid, and M. Rida, "A novel architecture combined with optimal parameters for back propagation neural networks applied to anomaly network intrusion detection," *Computers & Security*, 2018.

[7] H. Wang, J. Gu, and S. Wang, "An effective intrusion detection framework based on SVM with feature augmentation," *Knowledge-Based Systems*, vol. 136, pp. 130-139, 2017.

[8] M. S. M. Pozi, M. N. Sulaiman, N. Mustapha, and T. Perumal, "Improving anomalous rare attack detection rate for intrusion detection system using support vector machine and genetic programming," *Neural Processing Letters*, vol. 44, no. 2, pp. 279-290, 2016.

[9] P. Tao, Z. Sun, and Z. Suna, "An improved intrusion detection algorithm based on GA and SVM," *IEEE Access*, 2018.

[10] M. G. Raman, N. Somu, K. Kirthivasan, R. Liscano, and V. S. Sriram, "An efficient intrusion detection system based on hypergraph-Genetic algorithm for parameter optimization and feature selection in support vector machine," *Knowledge-Based Systems*, vol. 134, pp. 1-12, 2017.

[11] W. Shang, P. Zeng, M. Wan, L. Li, and P. An, "Intrusion detection algorithm based on OCSVM in industrial control system," *Security and Communication Networks*, vol. 9, no. 10, pp. 1040-1049, 2016.

[12] Y. Chen, Y. Li, X.-Q. Cheng, and L. Guo, "Building Efficient Intrusion Detection Model Based on Principal Component Analysis and C4. 5," in *Communication Technology, 2006. ICCT'06. International Conference on*, 2006, pp. 1-4: IEEE.

[13] G. Kim, S. Lee, and S. Kim, "A novel hybrid intrusion detection method integrating anomaly detection with misuse detection," *Expert Systems with Applications,* vol. 41, no. 4, pp. 1690-1700, 2014.

[14] D. Moon, S. B. Pan, and I. Kim, "Host-based intrusion detection system for secure human-centric computing," *The Journal of Supercomputing,* vol. 72, no. 7, pp. 2520-2536, 2016.

[15] D. Papamartzivanos, F. G. Mármol, and G. Kambourakis, "Dendron: Genetic trees driven rule induction for network intrusion detection systems," *Future Generation Computer Systems,* vol. 79, pp. 558-574, 2018.

[16] L. Prema Rajeswari and A. Kannan, "An intrusion detection system based on multiple level hybrid classifier using enhanced C4. 5," *Communications and Networking Madras Institute of Technology. Chennai, India: IEEE,* pp. 75-79, 2008.

[17] F. Y. Nia and M. Khalili, "An efficient modeling algorithm for intrusion detection systems using C5. 0 and Bayesian Network structures," in *Knowledge-Based Engineering and Innovation (KBEI), 2015 2nd International Conference on*, 2015, pp. 1117-1123: IEEE.

[18] O. Koucham, T. Rachidi, and N. Assem, "Host intrusion detection using system call argument-based clustering combined with Bayesian classification," in *SAI Intelligent Systems Conference (IntelliSys), 2015*, 2015, pp. 1010-1016: IEEE.

[19] H. Altwaijry, "Bayesian based intrusion detection system," in *IAENG Transactions on Engineering Technologies*: Springer, 2013, pp. 29-44.

[20] L. Xiao, Y. Chen, and C. K. Chang, "Bayesian model averaging of bayesian network classifiers for intrusion detection," in *Computer Software and Applications Conference Workshops (COMPSACW), 2014 IEEE 38th International*, 2014, pp. 128-133: IEEE.

[21] N. Devarakonda, S. Pamidi, V. V. Kumari, and A. Govardhan, "Intrusion detection system using bayesian network and hidden markov model," *Procedia Technology,* vol. 4, pp. 506-514, 2012.

[22] Z. Yi and Z. Li-Jun, "A rule generation model using S-PSO for Misuse Intrusion Detection," in *Computer Application and System Modeling (ICCASM), 2010 International Conference on*, 2010, vol. 3, pp. V3-418-V3-423: IEEE.

[23] I. Aljarah and S. A. Ludwig, "Mapreduce intrusion detection system based on a particle swarm optimization clustering algorithm," in *Evolutionary Computation (CEC), 2013 IEEE Congress on*, 2013, pp. 955-962: IEEE.

[24] Y. Y. Chung and N. Wahid, "A hybrid network intrusion detection system using simplified swarm optimization (SSO)," *Applied Soft Computing,* vol. 12, no. 9, pp. 3014-3022, 2012.

[25] A. S. Eesa, Z. Orman, and A. M. A. Brifcani, "A novel feature-selection approach based on the cuttlefish optimization algorithm for intrusion detection systems," *Expert Systems with Applications,* vol. 42, no. 5, pp. 2670-2679, 2015.

[26] C.-M. Ou, "Host-based intrusion detection systems adapted from agent-based artificial immune systems," *Neurocomputing,* vol. 88, pp. 78-86, 2012.

[27] A. H. Hamamoto, L. F. Carvalho, L. D. H. Sampaio, T. Abrão, and M. L. Proença Jr, "Network Anomaly Detection System using Genetic Algorithm and Fuzzy Logic," *Expert Systems with Applications,* vol. 92, pp. 390-402, 2018.

[28] M. Ahmed, A. N. Mahmood, and J. Hu, "A survey of network anomaly detection techniques," *Journal of Network and Computer Applications,* vol. 60, pp. 19-31, 2016.

[29] G. P. Gupta and M. Kulariya, "A framework for fast and efficient cyber security network intrusion detection using apache spark," *Procedia Computer Science,* vol. 93, pp. 824-831, 2016.

[30] L. P. Rajeswari and A. Kannan, "An Intrusion Detection System based on multiple level hybrid classifier using enhanced C4. 5," in *Signal Processing, Communications and Networking, 2008. ICSCN'08. International Conference on*, 2008, pp. 75-79: IEEE.

[31] B. Luo and J. Xia, "A novel intrusion detection system based on feature generation with visualization strategy," *Expert Systems with Applications,* vol. 41, no. 9, pp. 4139-4147, 2014.

[32] H. Wang, Y. Xiao, and Y. Long, "Research of intrusion detection algorithm based on parallel SVM on spark," in *Electronics Information and Emergency Communication (ICEIEC), 2017 7th IEEE International Conference on*, 2017, pp. 153-156: IEEE.

[33] S. Paliwal and R. Gupta, "Denial-of-service, probing & remote to user (R2L) attack detection using genetic algorithm," *International Journal of Computer Applications,* vol. 60, no. 19, pp. 57-62, 2012.