Recommendation based on guided analytics for Product prediction in retail space

J C Achutha
Assistant Professor
Department of Master Of Computer Applications
The Oxford College of engineering Bangalore

Abstract— Production and Data Reliability based on Cyber-attacks are becoming an increasing threat to people and daily businesses regularly. Attackers have also been evolving their strategies and methods with time. Every attack carried out has the potential to exploit the system on a large scale. Various Artificial Intelligence (AI) algorithms are used to defend such vulnerabilities. This paper analyzes a novel attack and extracts attackers' intrusion scenarios. Evolutionary Computation Techniques have been remarkably used in the field of cybersecurity. This paper particularly discusses the Distributed Denial Of Service (DDoS) attack. The effect of this attack ranges from a disturbance of an elementary service to causing major threats to critical services. In recent times these attacks have become more intricate and carry a significant threat. Therefore, there is a necessity for an intelligent Intrusion Detection System (IDS) to recognize attacks. In this study, work is carried on the latest dataset called Modern DDoS. This paper comprises of comparing the results of six established classification techniques: Random Forest, Naive Bayes, Stochastic Gradient Descent, Deci-sion Trees, Logistic Regression, and K-Nearest Neighbour (KNN) with the proposed Genetic Programming model. The results show that the proposed Genetic Programming model has better accuracy when compared to various existing methods.

Keywords—Intrusion Detection System (IDS), Distributed De-nial of Service (DDoS), Modern DDoS dataset, Evolutionary Computation (EC), Genetic programming (GP), Principal Com-ponent Analysis (PCA).

I. INTRODUCTION

Cybersecurity is becoming a regular struggle for organiza-tion asset's and businesses. The effort to hinder the integrity, confidentiality, or availability of system is called Intrusions. [3] "Intrusion Detection is the process of investigating and moni-toring the events occurring in the network or computer system which are violations or impeding threats of computer security policies" [5]. An Intrusion detection system (IDS) is an ap-plication that defends your network from suspicious activities, threats, and vulnerabilities when detected [7]. However, IDS faces several issues such as unbalanced data distributions, large traffic volumes, continuously changing environments and the need to recognize normal and abnormal behavior [14].

A Cyber attack is a deliberate attempt that targets one or more computers against multiple computers or networks. The Cyber Attacks such as Denial-of-service (DoS) and Distributed Denial-of-service (DDoS), Man-in-the-Middle (MitM) attack, Phishing, and Spear-Phishing attacks, Cross-site Scripting (XSS) attack, Malware attack, etc have attracted the attention of researchers over the years [16]. The primary focus of this study is particularly restricted to DDoS attacks which will be extended to various attacks in future and thus coming up with a model capable of detecting attacks of various kinds and providing an immediate remedy of the attack in case attack happens. A DDoS attack is a pernicious attack on network wherein the targeted system (a server or website or any other network resources) gets affected by causing the denial of services to the user of the targeted system (or resources) [23]. Hackers make use of botnets to flood an IP address with thousands of messages and connection requests thereby denying services to legitimate users.

Advances in Machine Learning (ML) and Deep Learning (DL) [24], [25] have a profound impact on science and tech-nology. These technologies have many recent successes in the field of Cyber-security. The study focuses on the usage of ML and Evolutionary Computation (EC) algorithms specifically Genetic Programming to investigate the IDS building process more effectively than the existing methods [26]. "Genetic programming (GP) is an evolutionary approach towards com-puting that focuses on optimal classification. GP is a meta-heuristic approach that is capable of using complex pattern representations such as trees" [27]. This paper demonstrates a comprehensive analysis of detecting DDoS attacks using various classification models as well as the proposed method using genetic programming. Within this evaluation, six ML models namely Random Forest, Naive Bayes, Stochastic Gra-dient Descent, Decision Trees, Logistic Regression, K-Nearest Neighbour (KNN), [28] and genetic programming model are explored for detecting DDoS attacks and their performances are evaluated based on experiments on Modern DDoS dataset.

The rest of the paper is organised as follows: Section II encapsulates the available literature. Section III sums up the genetic programming fundamentals and explanation. In section IV, the proposed method is reported in detail. Section V furnishes the experiments and results. Section VI bestows conclusions and future scope of the work.

II. Related Work

Espejo et.al [1] have surveyed how Genetic Programming can be used for classification. They have spoken of different methods of constructing a classifier which can be more ac-curate and dependable. The main aim was at upgrading the quality of classification by using GP. The distinctive elements of GP make it a dependabl technique for classification. It was concluded from this paper that different classification models such as Decision trees, Random Forest etc can be used as individuals of a population. Drawbacks of GP were also highlighted.

A literature survey of Machine Learning (ML) and Data Mining (DM) methods used for intrusion detection is portrayed by authors Buczakk and Erhan Guven [2]. Though they have discussed different ML/DL techniques, it is difficult to conclude which method is most efficient. There are various parameters from which the effectiveness of a model can be calculated since it depends on the particular IDS. They have discussed how datasets play a major role in training and testing models in the cyber intrusion.

Since there were no common datasets that contain new types of DDoS attacks, hence a new dataset was collected by Alka-sassbeh et.al [4]. The collected dataset was named as Modern DDoS. It composed of five DDos traffic classes. No redundant or duplicate records were found. Various methods such as collection and audition, preprocessing, feature extraction, and statistical measurements were performed before obtaining the dataset. Three established classification techniques were used for example Na¨ıve Bayes, Random Forest, and Multilayer Perceptron (MLP). Improved results over this paper have been discussed ahead in the comparative study.

Alyasiri et.al [5] have discussed a graph-based optimal approach for Genetic Programming called Cartesian Genetic Programming. Rules are constructed for the detection of different kinds of cyber attacks using this technique. The Modern DDoS dataset was used for experimentation. The Java Evolutionary Computation Toolkit (ECJ) was used for implementation. Suitable parameters such as population size, generations, mutation rate, etc were used while performing the experiments. The results of this approach are compared to the proposed GP model. There is a significant improvement in the results.

Mukkamala S.et.al [8] explored the feasibility of the Linear Genetic Programming (LGP) technique to model systematic IDS. Through a variety of experimentations, they have dis-cussed appropriate parameters such as program size, popula-tion size crossover rate, and mutation rate and proved in terms of accuracy that LGP programs can outrange Support Vector Machine and Artificial Neural Networks.

In [10] Ahvanooey et.al provided a comprehensive review of various aspects of Genetic Programming including key steps, selection strategies like a tournament, rank-based, ex-ponential, and truncation selection, crossover operators like single-point,n-point, uniform and flat crossover and mutation operators, and its applications in different scientific fields. It also aimed at providing an easy understanding of various types of GP including linear, grammatical evolution, cartesian, extended compact, probabilistic incremental program evolu-tion, and strongly-typed genetic programming along with their advantages and disadvantages.

Husak´ et.al [11] surveyed attack prediction, intention identi-fication, intrusion prediction, and network security forecasting. Three important conclusions from the survey were: The use of discrete models were used for attack projection and continuous models was used for forecasting. The dependence on artificial prediction models was resolved by Data mining. Problems were encountered relating to the analysis of forecasting in cybersecurity.

Al Najada et.al [12] presented a taxonomy for different types of attacks using Deep Learning. Forecasting models were created for each attack independently and then a forecasting model was created for all the attacks using deep learning and distributed random forest considering only a set of attributes to improve the accuracy. The class imbalance case was resolved using the oversampling technique. Their developed model could accurately forecast the type of attack or menace.

Yusof et.al [17] have presented a comprehensive systematic literature review on DDoS impact, which includes the defi-nition of DDoS attack, various types of DDoS attacks, the existing DDoS detecting techniques, and different kinds of prediction techniques. The result of their observation showed that the machine learning technique was significantly used in the prediction and detection of DDoS attacks.

III. Genetic Programming

Genetic Programming (GP) can be considered as an ex-tension of Genetic algorithms where one of the major dif-ference lies in consideration of initial population. The initial population in the genetic programming are computer programs which undergo selection and fitness function evaluation and further crossover operators and mutation are applied. GP was introduced by John Koza [22] as a type of Evolutionary Algorithm (EA) which evolves over time and hence solution becomes better over generations [22]. It is a method that procreates a population which consists of computer programs that solve a particular problem. They can be enhanced by using certain naturally occurring genetic operations. These programs are constructed using functions and there are a certain set of rules according to which they are executed [29]. These operations are performed iteratively until a better result is obtained. GP has the capability to evolve its problem space and problem representation to perceive regularity in different domains [29] [15] .

The execution steps of GP are shown in Fig. 1.

Initial Population is considered, it consists of various pro-grams or strategies depending on the problem. Not all the programs are optimal, hence each individual has a value given to it which is called as a fitness measure. This value can be in a numerical form which tells us how well the particular program performs. After applying suitable fitness measures, selection of these individuals is done using various methods which include

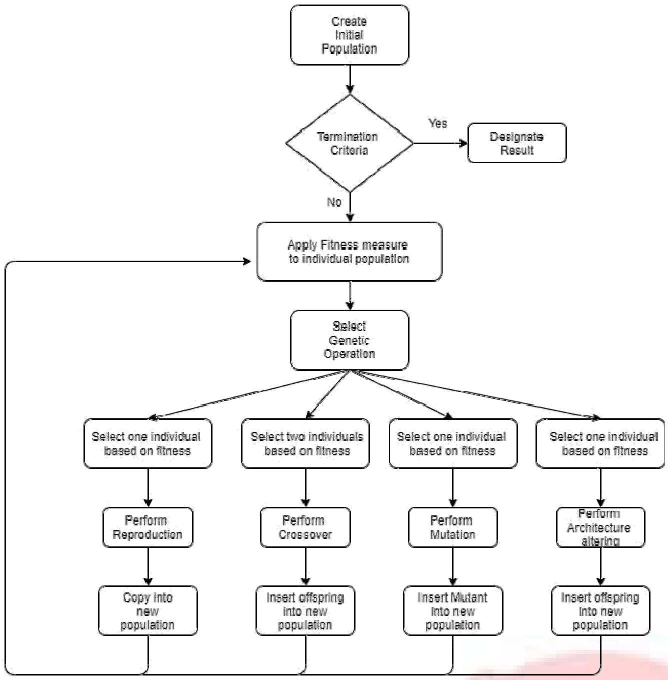

Fig. 1. Steps of Genetic Programming.

TABLE I
MODERN DDOS DATASET FEATURES [3]

Sr.no.	Attribute Name	Description
1	SRC_ADD	Source Address
2	DES_ADD	Destination Address
3	PKT_ID	Packet Identifier
4	FROM_NODE	Source Node
5	TO_NODE	Destination Node
6	PKT_TYPE	Packet Type
7	PKT_SIZE	Total Packet Size in Bytes
8	FLAGS	Flags
9	FID	Flag Identifier
10	SEQ NUMBER	Sequence Number
11	NUMBER_OF_PACKET	Total Number of Packets
12	NUMBER_OF_BYTE	Toatl Number of Bytes
13	NODE_NAME_FROM	Node Name From
14	NODE_NAME_TO	Node Name To
15	PKT_IN	Total time of packet inside queue
16	PKT_OUT	Total time of packet outside queue
17	PKT_R	Time of packet received
18	PKT_DELAY_NODE	Total packet delay within Node
19	PKT_RATE	Average packet rate
20	BYTE_RATE	Average byte rate
21	PKT_AVG_SIZE	Average packet size
22	UTILIZATION	Bandwidth utilization
23	PKT_DELAY	Total time packet delay
24	PKT SEND TIME	Time of sending packet
25	PKT RESERVED TIME	Time of receiving packet
26	FIRST PKT SENT	Time of first packet sent
27	LAST PKT RESERVED	Time of last packet received

Select Random, Select Best, Select Worst, Select Tournament, Select Roulette, Select Double Tournament, Select Stochastic Universal Sampling, etc. Once a suitable individual is selected four different operations can be performed: Reproduction, Crossover, Mutation, and Architecture Altering [30]. One individual is selected on which reproduction is performed. The new individual obtained is added to the initial population. Sim-ilarly, the crossover is performed by selecting two individual and the new offspring is added in the initial population. In mutation, a single parent is selected and mutated. The mutated individual obtained is added to the initial population based on its fitness value. The process is carried out in loop iteratively and is terminated by using certain criteria.

IV. PROPOSED METHOD

The proposed method captivates the following: (A) Data Acquisition, (B) Preparing data for further processing, and (C) Implementation of genetic programming for optimal results.

A. Data Acquisition

A novel dataset that contains modern kinds of DDoS attacks is used for this study. The Modern DDoS Dataset was generated using NS2 (Network Simulator) [4]. The dataset had 2,160,668 number of instances. The features of this dataset are listed in Table I. The distribution of Modern DDoS Dataset classes is comprised of Smurf, User Datagram Protocol-Flood (UDP-Flood), SQL Injection DDOS (SIDDOS), HTTP-Flood and Normal consisting of 12590, 201344, 6665, 4110 and 1935959 records respectively [3].

1. Smurf forwards a ping to a broadcast address using a spoofed source IP address. The target server receives a huge number of ICMP echo request packets. The victim machine is brought down when a large number of ICMP responses are forwarded.

2. User Datagram Protocol (UDP) flood a massive volume of UDP traffic is sent to inundate the chosen server, which leads the server passive to other clients.

3. SQL Injection DDOS (SIDDOS) a malicious code element usually an SQL statement is forwarded from client-side and sent to sever-side database.

4. HTTP flood is an attack where attackers overwhelm a server or application with authorized HTTP GET or POST requests. They wear-out the server resources responding to every request by acting as a legitimate user requesting services.

5. Normal transaction data.

The study in this paper is focused on reducing the complex-ity of the GP algorithm by not processing symbolic features such as Flags, Node Name From and Node Name To which are shown in Table I. In Packet Class feature, Smurf, UDP-Flood, SIDDOS, HTTP Flood are labelled as 1 and Normal is labelled as 0. Packet Type feature consists of four packets namely tcp, cbr, ack and ping which are labelled as 1, 2, 3 and 4 respectively.

B. Preprocessing

Principal Component Analysis (PCA) is an unsupervised dimensionality reduction technique that captures the maximum amount of variation in the data and finds principal components that are linear amalgams of initial attributes and that are orthogonal to each other [28].

PCA was imported from Scikit-learn [31], fit.transform function was used to train and test data. After applying PCA on the Modern DDoS Dataset the features are reduced to 8,16 and 20 principle components as shown in Table II. Though other numbers of principle components were also explored to study the loss of information and 8,16 and 20 were chosen based on the percentage of information loss. Since there is no significant difference in the 16 and 20 principal components in terms of information loss, 16 principal components were considered for simplicity for further processing.

TABLE II
PCA RESULTS

No. of Principal Components	% Information Gain	%Information Loss
8	94.92%	5.08%
16	98.48%	1.52%
20	99.6%	0.4%

C. Implementation of Genetic Programming (GP)

For implementation, Distributed Evolutionary Algorithm (DEAP) framework is used which is built over Python programming language. It provides necessary elements for creating sophisticated evolutionary computing systems. The implementation of GP is performed in four steps. The first step is to build an appropriate type of problem in this case a GP type is built. This is done using the creator module. A runtime creation of classes is performed using Creator module through inheritance and composition. Creator function consists of three parameters: name, base, and attribute. Attributes are dynamically added to the existing classes because of which creation of population is possible from any data-structure such as lists, sets, dictionaries, trees, etc. The second step is creating a fitness class using the creator module. The fitness of each individual is computed and the best individual is used for the next iteration. The third step is the initialization of operators in which the 'toolbox' module is used. The toolbox is a collection of operators. In the proposed model, the crossover operator used is single-point crossover hence the parameter passed into the toolbox is "cxOnePoint" Similarly, mutation operation is carried out using node replacement passing "mutNodeReplace-ment" as a parameter. Selection is performed using Double Tournament selection passing "selDoubleTournament" as a parameter. The final step consists of constructing the main function of the model in which the crossover rate, mutation rate, and the number of generations are set. This algorithm is terminated when the iterations of all the generations are completed. Fig. 2 shows the execution of steps carried out during implementation.

V. EXPERIMENTS AND RESULTS

All the experiments and implementations were performed on Intel Core i7-8550U CPU Processor, 16GB RAM and 64-bit Operating system. Softwares used were Spyder and Jupyter Notebook.

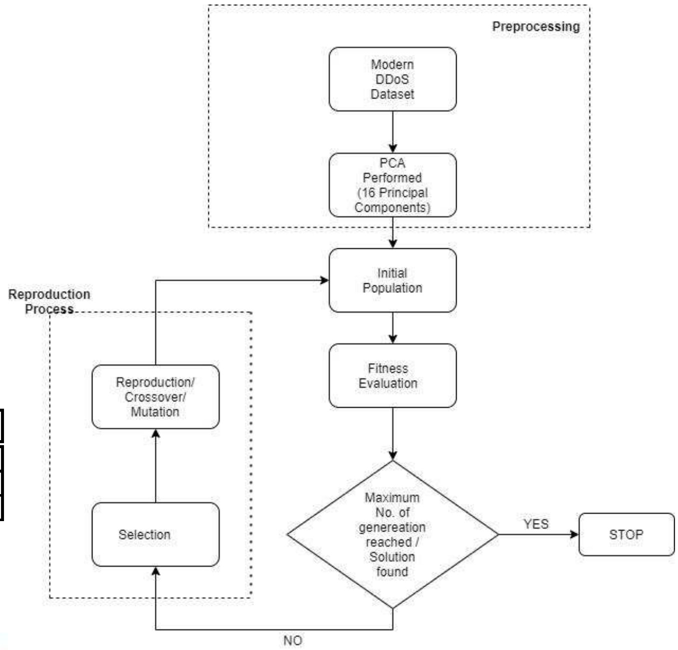

Fig. 2. Flow chart of Process.

DEAP [21], a novel evolutionary computation framework and TPOT [18], [19], [20], a tree based optimization tool is used for GP implementation. Table III summarizes the comparison of accuracy of various existing intrusion detection systems and our proposed system.

The Modern DDoS dataset which is a supervised dataset, is used for the experimentation. The accuracy of the proposed algorithm can be evaluated in such a way that it should tell how malicious and normal behaviours are classified. The Modern DDoS dataset originally is having class labels as SMURF, UDP-Flood, SIDDOS, HTTP Flood, and Normal. Amongst these as stated above, the first four are malicious DDoS attacks and Normal indicates no attack. So, these four attacks (class labels) are replaced by 1 and Normal by 0 to bring simplicity as the current study targets only at detecting an attack or no attack.

The confusion matrix is the most widely adopted statistical measure for binary classification problems. A confusion matrix consists of: True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). There are some derived measurements that are [3]:

$$DetectionRate(DR) = \frac{TP}{TP+FN} \quad (1)$$

$$Accuracy = \frac{TP+TN}{TN+TP+FN+FP} \quad (2)$$

$$FalsePositiveRate(FPR) = \frac{FP}{FP+TN} \quad (3)$$

$$FalseNegativeRate(FNR) = \frac{FN}{FN+TP} \quad (4)$$

TABLE III
COMPARISON OF VARIOUS MODELS AND PROPOSED APPROACH

Author	Year	Model	Dataset	Accuracy
S. Umarani, D.Sharmila [6]	2015	Naive Bayes	1998 World Cup Website	95.95%
Naveen Bindra, Manu Sood [9]	2019	Random Forest	CIC IDS 2017	96.13%
Mo.Alkasassbeh et.al [4]	2016	Naive Bayes	Modern DDoS	96.91%
Manjula Suresh, R. Anitha [13]	2011	Naive Bayes	CAIDA	97.20%
Hasanen Alyasiri et.al [5]	2018	Genetic Programming: Cartesian	Modern DDoS	97.19%
Mo.Alkasassbeh et.al [4]	2016	Random Forest	Modern DDoS	98.02%
Proposed Model	2020	Genetic Programming	Modern DDoS	98.67%

$$F\,alseAlarmRate(F\,AR) = \frac{FPR + FNR}{2} \quad (5)$$

TP indicates the cases that are correctly classified as an attack or malicious behavior, TN indicates the cases that are correctly classified as normal behavior or no attack. FN indicates the cases that are incorrectly classified as malicious behavior and FP indicates the cases that are incorrectly clas-sified as normal behavior, both of these being problematic. Eq.1 specifies the fraction of cases that are correctly classified as a malicious attack. Eq.2 describes the fraction of correctly predicted attacks to all attacks or non-attacks that are correctly classified. Eq.3 defines normal behaviors incorrectly classified as malicious. Eq.4 defines malicious behaviour that are er-roneously predicted as normal behavior. Eq.5 calculates the wrongly classified attacks [3].

The confusion matrix values of the respective models are shown in Table IV. Table V portrays the accuracy results of different classification models which are implemented using Modern DDoS supervised Dataset.

TABLE IV
CONFUSION MATRIX DETAILS OF VARIOUS CLASSIFIERS

Model	TP	TN	FP	FN
KNN	387179	38985	127	5843
Naive Bayes	378513	39200	8793	5628
Logistic Regression	387181	38987	125	5841
Decision Tree	381621	39125	5685	5703
Random Forest	384690	39099	2616	5729
Stochastic Gradient Descent	386897	38976	409	5852

TABLE V
ACCURACY RESULTS OF CLASSIFICATION MODELS

Model	DR	FAR	Accuracy
KNN	98.51	0.09	98.57
Naive Bayes	98.53	9.88	96.66
Logistic Regression	98.52	0.09	98.62
Decision Tree	98.52	7.09	97.38
Random Forest	98.53	3.86	98.07
Stochastic Gradient Descent	98.50	7.08	98.55

The GP implementation depends on various parameters such as the population size, number of generations, crossover rate, mutation rate, verbosity etc. After passing suitable values to the parameters it was observed that when the population size was 50, by passing a crossover rate of 0.01 an accuracy of 98.67% was obtained. Fig. 3 shows the relationship between the accuracy and population size of the proposed model. As the population size increases the accuracy is stabalizing towards 98.66%. Fig. 4 shows that as the number of generations increases by a significant amount, the accuracy did not change much.

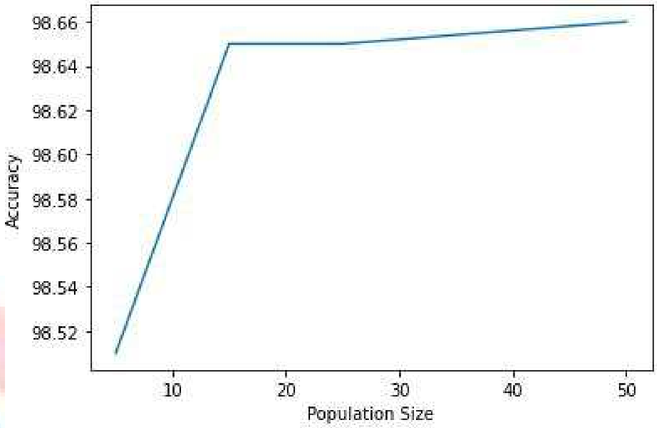

Fig. 3. Accuracy VS Population Size of Proposed Model.

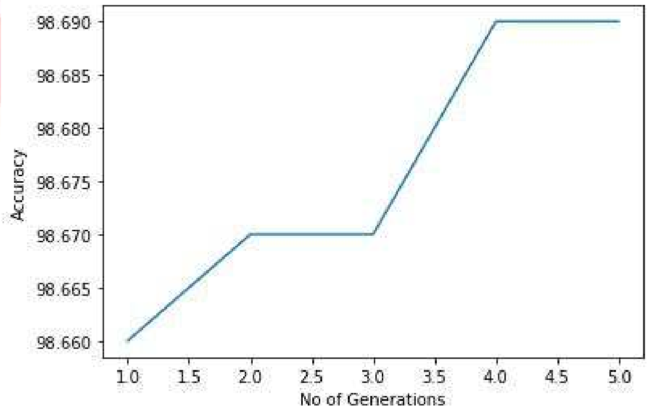

Fig. 4. Accuracy vs No. of Generations of Proposed Model.

VI. CONCLUSION

This scientific analysis investigates an application of Ge-netic Programming (GP) for intrusion detection. For this study, the Modern DDoS dataset is used. This dataset contains contemporary threats gathered from various environments. The proposed GP model detects DDoS attacks with improved accuracy of 98.67% while comparing it with six established classification models. The obtained results highlight the advan-tages of adopting the GP model. However, it was observed that adopting other approaches for operations such as mutation or crossover can result in better results. Due to limited resources, this was not tested. In future, this model can be investigated for other types of attacks and also to come up with a universal model to detect all kinds of well-known threats.

REFERENCES

[1] Espejo, Pedro G., Sebastian´ Ventura, and Francisco Herrera. ”A survey on the application of genetic programming to classification.” IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 40, no. 2 (2009): 121-144

[2] Buczak, Anna L., and Erhan Guven. ”A survey of data mining and machine learning methods for cyber security intrusion detection.” IEEE Communications surveys tutorials 18, no. 2 (2015): 1153-1176.

[3] Alyasiri, Hasanen. ”Developing Efficient and Effective Intrusion Detec-tion System using Evolutionary Computation.” PhD diss., University of York, 2018.

[4] Alkasassbeh, Mouhammd, Ghazi Al-Naymat, Ahmad Hassanat, and Mo-hammad Almseidin. ”Detecting distributed denial of service attacks using data mining techniques.” International Journal of Advanced Computer Science and Applications 7, no. 1 (2016): 436-445.

[5] Alyasiri, Hasanen, John A. Clark, and Daniel Kudenko. ”Applying Cartesian Genetic Programming to Evolve Rules for Intrusion Detection System.” In IJCCI, pp. 176-183. 2018.

[6] Umarani, S., and D. Sharmila. ”Predicting application layer DDoS attacks using machine learning algorithms.” International Journal of Computer and Systems Engineering 8, no. 10 (2015): 1912-1917.

[7] Bace, Rebecca Gurley. Intrusion detection. Sams Publishing, 2000.

[8] Mukkamala S., Sung A.H., Abraham A. (2004) Modeling Intrusion Detection Systems Using Linear Genetic Programming Approach. In: Orchard B., Yang C., Ali M. (eds) Innovations in Applied Artificial Intelligence. IEA/AIE 2004. Lecture Notes in Computer Science, vol 3029. Springer, Berlin, Heidelberg

[9] Bindra, Naveen, and Manu Sood. ”Detecting DDoS attacks using ma-chine learning techniques and contemporary intrusion detection dataset.” Automatic Control and Computer Sciences 53, no. 5 (2019): 419-428.

[10] Ahvanooey, Milad Taleby, Qianmu Li, Ming Wu, and Shuo Wang. ”A Survey of Genetic Programming and Its Applications.” TIIS 13, no. 4 (2019): 1765-1794.

[11] Husak,´ Martin, Jana Komarkov´a,´ Elias Bou-Harb, and Pavel Celeda. ”Survey of attack projection, prediction, and forecasting in cyber secu-rity.” IEEE Communications Surveys Tutorials 21, no. 1 (2018): 640-660.

[12] Al Najada, Hamzah, Imad Mahgoub, and Imran Mohammed. ”Cyber Intrusion Prediction and Taxonomy System Using Deep Learning And Distributed Big Data Processing.” In 2018 IEEE Symposium Series on Computational Intelligence (SSCI), pp. 631-638. IEEE, 2018.

[13] Suresh, Manjula, and R. Anitha. ”Evaluating machine learning al-gorithms for detecting DDoS attacks.” In International Conference on Network Security and Applications, pp. 441-452. Springer, Berlin, Hei-delberg, 2011.

[14] Wu, Shelly Xiaonan, and Wolfgang Banzhaf. ”The use of computational intelligence in intrusion detection systems: A review.” Applied soft computing 10, no. 1 (2010): 1-35.

[15] Miller, Julian Francis. ”Cartesian genetic programming: its status and future.” Genetic Programming and Evolvable Machines (2019): 1-40.

[16] Akbari Roumani, M., Chun Che Fung, Shri Rai, and Hong Xie. ”Value analysis of cyber security based on attack types.” ITMSOC: Transactions on Innovation and Business Engineering 1 (2016): 34-39.

[17] Yusof, Ahmad Riza’ain, Nur Izura Udzir, and Ali Selamat. ”Systematic literature review and taxonomy for DDoS attack detection and prediction.” International Journal of Digital Enterprise Technology 1, no. 3 (2019): 292-315.

[18] Le, Trang T., Weixuan Fu, and Jason H. Moore. ”Scaling tree-based automated machine learning to biomedical big data with a feature set selector.” Bioinformatics 36, no. 1 (2020): 250-256.

[19] Olson, Randal S., Ryan J. Urbanowicz, Peter C. Andrews, Nicole A. Lavender, and Jason H. Moore. ”Automating biomedical data science through tree-based pipeline optimization.” In European Conference on the Applications of Evolutionary Computation, pp. 123-137. Springer, Cham, 2016.

[20] Olson, Randal S., Nathan Bartley, Ryan J. Urbanowicz, and Jason H. Moore. ”Evaluation of a tree-based pipeline optimization tool for automating data science.” In Proceedings of the Genetic and Evolutionary Computation Conference 2016, pp. 485-492. 2016.

[21] Fortin, Felix´-Antoine, Franc¸ois-Michel De Rainville, Marc-Andre´ Gard-ner Gardner, Marc Parizeau, and Christian Gagne´. ”DEAP: Evolutionary algorithms made easy.” The Journal of Machine Learning Research 13, no. 1 (2012): 2171-2175.

[22] Koza, John R., and John R. Koza. Genetic programming: on the programming of computers by means of natural selection. Vol. 1. MIT press, 1992.

[23] Douligeris, Christos, and Aikaterini Mitrokotsa. ”DDoS attacks and defense mechanisms: classification and state-of-the-art.” Computer Net-works 44, no. 5 (2004): 643-666.

[24] KP, Soman, and Mamoun Alazab. ”A Comprehensive Tutorial and Survey of Applications of Deep Learning for Cyber Security.” (2020).

[25] Xin, Yang, Lingshuang Kong, Zhi Liu, Yuling Chen, Yanmiao Li, Hongliang Zhu, Mingcheng Gao, Haixia Hou, and Chunhua Wang. ”Machine learning and deep learning methods for cybersecurity.” IEEE Access 6 (2018): 35365-35381.

[26] Al Najada, Hamzah, Imad Mahgoub, and Imran Mohammed. ”Cyber Intrusion Prediction and Taxonomy System Using Deep Learning And Distributed Big Data Processing.” In 2018 IEEE Symposium Series on Computational Intelligence (SSCI), pp. 631-638. IEEE, 2018.

[27] Koza, John R., and Riccardo Poli. ”Genetic programming.” In Search methodologies, pp. 127-164. Springer, Boston, MA, 2005.

[28] Tan, Pang-Ning, Michael Steinbach, and Vipin Kumar. Introduction to data mining. Pearson Education India, 2016.

[29] Hansen, James V., Paul Benjamin Lowry, Rayman D. Meservy, and Daniel M. McDonald. ”Genetic programming for prevention of cybert-errorism through dynamic and evolving intrusion detection.” Decision Support Systems 43, no. 4 (2007): 1362-1374.

[30] Nordin, Peter, Robert E. Keller, and Frank D. Francone. Genetic programming. Edited by Wolfgang Banzhaf. Springer, 1998.

[31] Pedregosa, Fabian, Gael¨ Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel et al. ”Scikit-learn: Machine learning in Python.” the Journal of machine Learning research 12 (2011): 2825-2830.