UtilitasMathematica

ISSN 0315-3681 Volume 122, 2025

Ensemble Learning for Detecting Application-Layer DDoS Attacks from Open-Source Toolkits

[1,] Nuthi Vinay Kumar, [2,] Jampani Satish Babu

Department of Computer Science and Engineering, KL University, (Vijayawada)

[1, 2,] vinaynuthi2001@gmail.com, jampanisatishbabu@kluniversity.in

Abstract: The aim of the project is to detect and alleviate DDoS Application-Layer escalating attacks and provide insight into patterns of attack and tools for increased cyber security measures. In order to attack HTTP-birds, the project seeks to detect tactics and tools and offer a specialized approach to strengthen understanding and countermeasures against developing cyber threats. It is urgently necessary to solve the growing threat of DDOS by shifting the project focus on the accessibility of tools. This is essential for active defense against the widespread use of harmful offensive equipment. The aim of the project is to seize the network administrator and cyber security experts and provide online services. Finally, it benefits users and businesses with durable defense against developing DDOS threats. "To boost performance, we introduced ensemble models—voting Classifier (RandomForest, DecisionTree) and Stacking Classifier (RandomForest, DecisionTree, LGBM)". The aim of these improvements is to improve the accuracy of cyberbully detection.

"Index terms - *DDoS, DDoS tools, machine learning, deep learning".*

1. INTRODUCTION

The aim of the project is to detect and stop rising attacks on the application layer. It will also provide information about attack patterns and tools for better cybersecurity. The project's goal is to figure out how to stop HTTP-layer assaults. Distributed denial of service (DDoS) attacks [1], [2] are one of the most harmful and complicated security threats to computer networks. There was a big rise in application-layer assaults in the first quarter of 2022. "HTTP-layer DDoS attacks, for example, went up 164% year over year and 135% quarter over quarter". The consumer Electronics sector had the most rise in industrial assaults, "with a shocking 5,086% QoQ. Online Media came in second with a 2,131% rise" in assaults from one quarter to the next, while pc software firms came in third with a 76% rise from one quarter to the next and a 1,472% rise from one year to the next [2].

DDOS attack is a bad attempt to stop a particular website, computer or network in the right job by flooding it by operation from many different sources. In this type of attack, the "Botnet" attacker, a network of computers or other devices, uses too much data to the target system, which makes it impossible for justified users to access it [4]. DDoS assaults are complicated because (1) they may create a lot of traffic from many different

places, and (2) the traffic seems to come from many different places [5].

There are many different types of DDoS "assaults, and application layer" attacks are one of them. The Application-Layer DDOS attacks are trying to reduce the resources of the victim's application or make an application crash. HTTP floods and Slowloris attacks are two examples of these types of attacks. The main purpose of DDOS attacks is to make the network useless by flooding it by operation, which can cause system failure or make it unavailable [6], [7] and [8].

The fact that DDoS attack tools are simple to get is a big reason why DDoS assaults are on the rise. These programs may be used on purpose to send so much traffic to servers and websites that they stop working. Because tools are so simple to get, "either by buying them on the dark web or downloading scripts that are free to use", individuals who don't know much about computers may carry out severe DDoS assaults [4], [9], [10], [11].

D technologies that take a unique approach to improving "knowledge and defenses" against new cyber threats. We need to quickly deal with the growing DDoS threats by making tools more accessible. This is very important for protecting yourself from the widespread usage of harmful attack tools. "The goal of the initiative is to provide network managers and cybersecurity professionals more control, which will make internet services safer". In the end, it helps both individuals and organizations have strong defenses against DDoS attacks that change over time. "We added voting Classifier (RandomForest, DecisionTree) and Stacking Classifier (RandomForest, DecisionTree, LGBM) to improve performance". The goal of these improvements is to make it easier to find cyberbullying.

2. LITERATURE SURVEY

IoT-based solutions are used in every sector of business, from smart homes to smart power grids to industrial automation. These devices made the attack surface bigger and were easier for the attacker to access to since they couldn't use heavy security measures because they didn't have the resources. IoT devices are less safe and usually work when no one is around, and therefore attackers have to build a botnet army to start a huge attack on the rejection of service [1, 2 and 17]. This study [1] speaks of a way to find the operation of an attack in consumer IoT (ciot) by machine learning. This method uses local features specific to IoT network to allow cheap classifiers ML to find attacks on the local router. The test results showed that the proposed method had the highest accuracy of 0.99, which shows that it is strong and reliable in IoT networks.

Today, the internet is becoming a new way to handle industrial applications from a distance, such power plant operators who need to control another site from a distance. "Denial-of-service (DoS) assaults may create major problems on the internet, which might put the functioning of network-based control systems at risk". Some people have suggested using overlay networks to secure internet application sites by disguising their location. This publication [2] looks at a lot of different ways that have been tried to solve this issue. This article talks about how to construct an interface for an overlay network that protects

communication services between application sites that are spread out over a wide area against DoS assaults. This work talks about a new architecture called "overlay protection layer (OPL) that protects application sites against DoS assaults before they happen [17, 18]". This research uses simulation to prove that DoS attacks are very unlikely to interfere with communication services via the OPL architecture. Even if attackers use a distributed DoS assault to take down half of the overlay nodes, 75% of the communication channels will still be open.

"In the closing several years, hackers have mostly used DDoS attacks". This is because it may cause a lot of different complications. Many hackers and specialists in this industry have developed packages and programs that begin with DDOS attacks on different types of networks. If you want to come up with effective ways to stop DDOS attacks, it is important to try and compare the power of attacks that can run these tools. "This research [4] compares and analyzes the performance of three DDoS attack tools based on things like how long it takes to successfully start an attack, the traffic rate, and the packet size". "[20] Slowloris, GoldenEye, and Xerxes are some of the DDoS tools that are being looked at". The findings of the trial suggest that Xerxes is better than other tools at starting a DDoS assault [21, 23].

"Software defined Networking (SDN) is a new kind of networking that separates control decisions from forwarding hardware". He claims to make network management and allows new ideas and growth. In SDN, software controllers (control plane) conceptually centralize network knowledge. Network devices (openflow switches) become basic packet transmission devices (data plane) that can be configured on an open interface (openflow protocol). "This separation of the control plane from the data plane creates a number of problems, such as security, dependability, load balancing, and traffic engineering". "Denial of service (DoS) and distributed denial of service (DDoS) attacks are of the worst security problems in SDNs [5]". "For example, with SDNs, DoS and DDoS assaults might fill up the control plane, the data plane, or the communication channel". Attack on the control plane can reduce the entire network while attacking a data plane or communication channel can cause packet loss and the network will not be available. In this study we provide a number of posts that help clarify DOS/DDOS attack domain in SDN. This includes a complete overview of the subject, including attacks and analysis of remedies already available [19, 20]. In short, our efforts may be summed up as follows: "We look at and organize the best ways to stop both DoS and DDoS assaults in SDNs from both an intrinsic and an extrinsic point of view". "Also, the countermeasures we spoke about are grouped by what they do, whether it's detection, mitigation, prevention, or gentle deteriorSation". We also look at the many ways and instruments that were used to put the new ideas into action. Lastly, we talk about prospective future research areas that might help with DoS/DDoS assaults in SDNs [21].

"DDoS (distributed denial-of-service)" is an issue that is becoming worse quickly. There are so many different kinds of assaults and defenses that it's hard to keep track of them all. This work [6] gives ways to group assaults and defenses, which helps

academics better comprehend the issue and the range of possible solutions. We chose the attack categorization criteria to show the similarities and key elements of attack techniques that create problems and determine how to construct responses. The defensive taxonomy sorts the current DDoS defenses into groups depending on the design choices that went into them. It then explains how these choices affect "the pros and cons of suggested solutions [4, 23]".

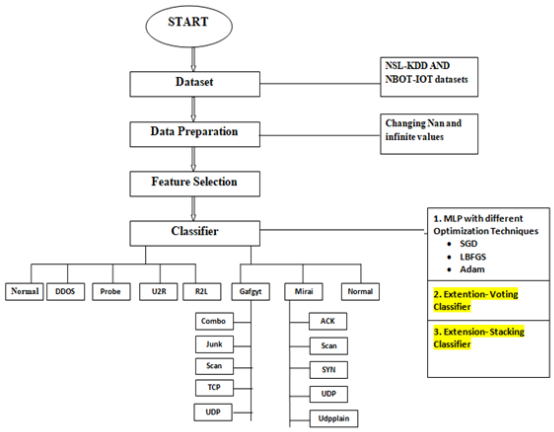
3. METHODOLOGY

i) Proposed Work:

The suggested technique for finding DDoS attacks goes beyond only seeing attack patterns [20]. It takes a more thorough approach. Instead, we look at the wider picture by looking at how easy it is to get at and how much damage readily accessible assault tools may do. This wider view helps us better comprehend cybersecurity risks and lets us build better defenses. The system may change to deal with new threats, so it doesn't just look at certain sorts of attacks. We added additional ensemble models to the project to improve performance. "These included a voting Classifier that combined RandomForest and DecisionTree, and a Stacking Classifier that used RandomForest and DecisionTree as foundation learners [20]". The goal of this ensemble method is to make it easier to find cyberbullying. "Also, a user-friendly Flask framework with SQLite was set up for secure registration and sign in". This made it easier for users to test the system by letting them enter data and get results. these enhancements not only make model designs more varied for better accuracy, but they also make it easier for users to engage with the project, which makes it more resilient and useful in real life.

ii) System Architecture:

The first step in the system design is to prepare the data using "the NSL-KDD [13] and NBOT-IOT datasets". Next, feature selection is done to make the data easier to analyze. After that, three classifiers are used: "MLP with different optimization methods (SGD, LBFGS, Adam), Extension Stacking Classifier, and Extension voting Classifier". These classifiers all work together to make predictions more accurate. The system's whole design makes it a flexible and powerful way to find and stop DDoS assaults in network security. It can analyze and forecast attacks "with a high degree of accuracy".



"Fig 1 Proposed architecture"

iii) Dataset collection:

"NSL-KDD Dataset":

NSL-KDD data file is a standard data file that many people use to test disturbance detection systems. This is a better version of the original KDD Cup 99

data file that solves its problems. "NSL-KDD has a wide range of network traffic data, including regular traffic and different kinds of assaults. This makes it a good choice for training and testing MLmodels in the area of cybersecurity [13]".

"NBOT-IOT Dataset":

"The NBOT-IOT dataset" is all about looking at how networks behave for the "internet of things (IoT)". It contains data from different IoT devices that might help us understand how devices communicate with each other and what security risks could exist in IoT networks. This dataset is very important for creating ML models that can find unusual behavior and possible security dangers that are unique to IoT settings.

	duration	protocol_type	service	flag	src_bytes	dst_bytes	land	wrong_fragment	urgent	hot	...	dst_host_srv_count	dst_host_same_srv_rate	dst_host_d
0	0	tcp	http	SF	181	5450	0	0	0	0	...	9	1.0	
1	0	tcp	http	SF	239	486	0	0	0	0	...	19	1.0	
2	0	tcp	http	SF	235	1337	0	0	0	0	...	29	1.0	
3	0	tcp	http	SF	219	1337	0	0	0	0	...	39	1.0	
4	0	tcp	http	SF	217	2032	0	0	0	0	...	49	1.0	

5 rows × 42 columns

"Fig 2 NSL KDD dataset"

iv) "Data Processing":

"Data processing is the process of turning raw data into useful information for organizations. Data scientists usually handle data by gathering it, organizing it, cleaning it, checking it, analyzing it, and turning it into graphs or texts that others can read". There are three ways to process data: "by hand, by machine, or by computer". The goal is to make information more useful and help people make decisions. This helps organizations run better and make critical choices on time. Automated data processing tools, like computer programming, are a big part of this. It can help you make sense of a lot of data, even huge data, so you can make better decisions and manage quality better.

v) Feature selection:

"Feature selection is the process of picking out the traits that are the most consistent, useful, and not repeated". As datasets become bigger and more diverse, it's vital to systematically make them smaller. The fundamental purpose of feature selection is to make a predictive model work better and cost less to run.

The selection of features is a key part of engineering. It is a process of choosing the most important characteristics that can feed into machine learning algorithms. Strategies of features are used to reduce the number of input variables by getting rid of functions that are not needed or are not useful and focus on those that are most important for the ML model. The key advantages of doing feature selection ahead of time instead than allowing the ML model find out which characteristics are most significant.

vi) "Algorithms":

The "**Multilayer Perceptron (MLP)** with the Stochastic Gradient Descent (SGD)" optimization technique is used in the project since it is good at training neural networks. SGD is a kind of gradient descent that picks a random batch of training data for each iteration, which makes it faster to compute. The combination of "MLP and SGD" works well for this project because it learns and adapts quickly to complicated patterns in the "NSL-

KDD and NBOT-IOT datasets". This makes it a strong base for DDoS attack detection since it can move across high-dimensional feature spaces [20].

"MLP combined with LIMITED-MEMORY BROYDEN-FLETCHER-GOLDFARB-SHANNO (LBFGS)": "Little memory Broyden-Fletcher-Goldfarb-Shanno (lbfgs) algorithm is a quasi-Newton optimization procedure that works on problems with no constraints". The LBFGS technique is a type of quasi-newton method that seeks to determine the minimum function without actually calculating its derivatives. LBFGS maintains the inverse of the inverse Hessian matrix to update the model parameters in steps. It works best when the dataset isn't too big and the model has a reasonable amount of parameters. If the dataset is not too big and the model has a good amount of parameters, lbfgs might work for THIS project. It usually converges quicker than certain other optimization techniques, particularly when the data fits well in memory.

"MLP – Adam": Adam (Adaptive moment Estimation) is a technique for optimizing the learning rate that uses ideas from both "momentum and RMSprop". It changes the learning rates of each parameter separately depending on their historical gradients. This makes it a good choice for situations with sparse gradients or noisy data. Many DL projects use Adam by default since it is recognized for being fast. It has ways to regulate both the size of the steps and the exponential decay of prior gradients [21].

Using a final estimator (LGBMClassifier), the "**Stacking Classifier"** aggregates predictions from basic classifiers like Random forest and decision Tree. It improves accuracy and resistance to changing DDoS assaults by using a variety of classifiers, which makes it a good fit for the project's network security aims.

The "**voting Classifier"** uses a "soft" voting system to combine the predictions of a "Random forest Classifier and a decision Tree Classifier" that have been refined using GridSearchCV. This ensemble method improves prediction effectiveness by taking into account weighted averages of class probabilities. This makes it useful for detecting DDoS attacks. Using a variety of classifiers makes defenses stronger against a wider range of cyber threats, making your security plan more effective and durable.

4. EXPERIMENTAL RESULTS

"Accuracy": The test accuracy is how well the difference between sick and healthy people can recognize. To find out how accurate the test is, we need to find out the percentage of real positives and real negatives in all cases they looked at. In terms of mathematics, this can be mentioned as:

$$"Accuracy = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{FP} + \mathrm{TN} + \mathrm{FN}} (1)"$$

"F1-Score": Score F1 is a way to check, how accurate the ML model is. It adds the accuracy of the model and the download score. Accuracy statistics will tell you how many times the model has created a valid forecast on the entire data file.

$$"F1\ Score = 2 * \frac{Recall\ \mathrm{X}\ Precision}{Recall + Precision} * 100(2)"$$

"Recall": Incalling is a ML statistics that shows how well the model can find all the relevant examples of a particular class. It is the ratio of properly predicted positive observations to the total number of real positives. This gives us an idea of how well the model captures examples of a particular class.

$$"Recall = \frac{\text{TP}}{\text{TP} + \text{FN}} (3)"$$

"Precision": Precision looks at the percentage of accurately categorized cases or samples that were labeled as positives. So, the formula for figuring out the accuracy is as follows:

$$"Precision = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} (4)"$$

In Table (1 & 2), check the performance metrics for each method, such as "accuracy, F1 score, recall, and precision". The Stacking Classifier always beats all the other algorithms on all criteria. The tables also show how the metrics for the different algorithms stack up against each other.

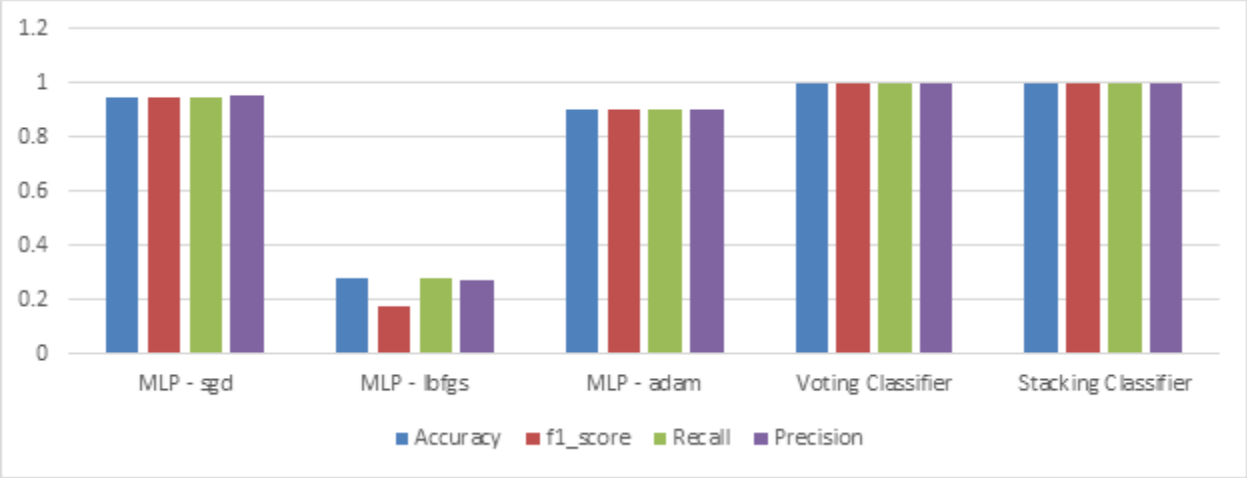
"Table.1 Performance Evaluation Table- NSLKDD DATASET"

ML Model	Accuracy	f1_score	Recall	Precision
MLP - sgd	0.943	0.947	0.943	0.953
MLP - lbfgs	0.282	0.175	0.282	0.269
MLP - adam	0.898	0.898	0.898	0.899
Voting Classifier	0.998	0.998	0.998	0.998
Stacking Classifier	**0.999**	**1.000**	**0.999**	**1.000**

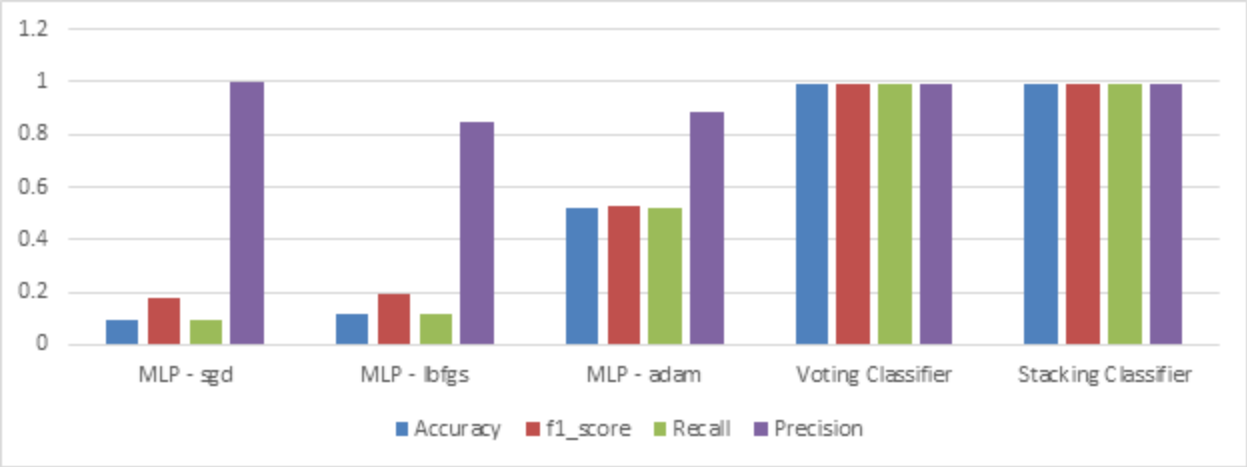
"Table.2 Performance Evaluation Table- NBOT-IOT DATASET"

ML Model	Accuracy	f1_score	Recall	Precision
MLP - sgd	0.098	0.178	0.098	1.000
MLP - lbfgs	0.117	0.191	0.117	0.844
MLP - adam	0.524	0.529	0.524	0.883
Voting Classifier	0.995	0.995	0.995	0.995
Stacking Classifier	**0.995**	**0.995**	**0.995**	**0.995**

"Graph.1 Comparison Graph- NSLKDD DATASET"

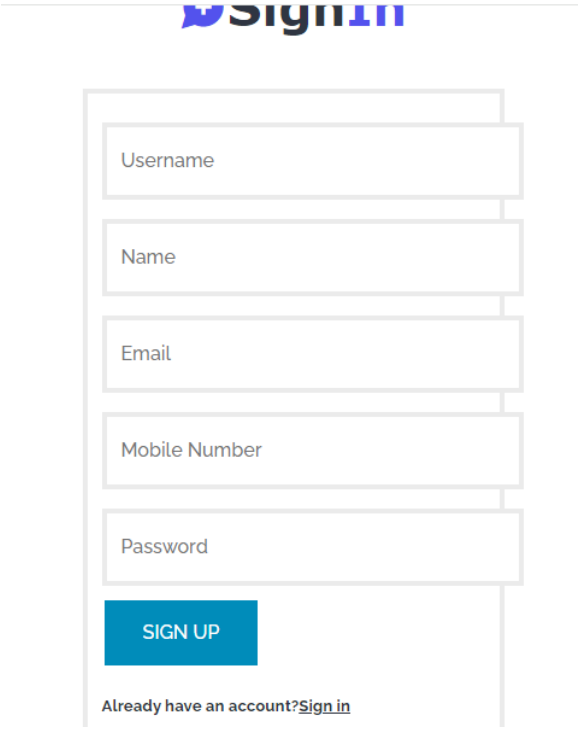


*"**Graph.2** Comparison Graph- NBOT-IOT DATASET"*



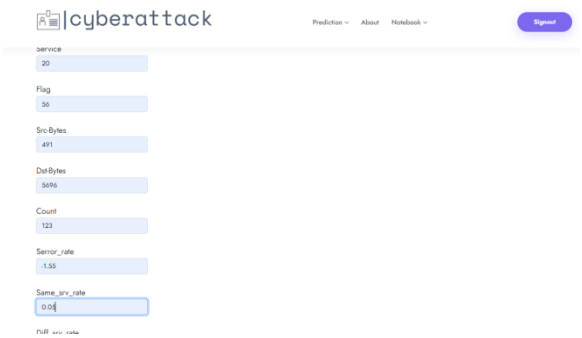
"Accuracy is represented in blue, F1 - Score in red, recall in green and precision in purple" ***Graph (1 & 2)***. The stacking classifier works better than other models on all measures and gets the highest score. The graphs above show these results in a clear way.

"Fig 3 Home page"

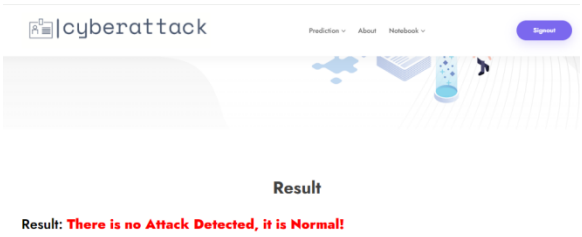


"Fig 4 Signin page"

"Fig 5 Login page"



"Fig 6 User input"



"Fig 7 Predict result for given input"

5. CONCLUSION

The research makes a big difference in cybersecurity by creating better ways to detect and stop "distributed Denial of service (DDoS)" assaults, which makes computer networks more resilient [16, 17]. By carefully looking at several datasets (KDD-CUP and NBOT-IOT), we learned a lot about how network traffic works and what kinds of attacks may happen. This gave us a good starting point for building successful models. "The examination of Multi-Layer Perceptron (MLP) [21] models with various optimizers, including as SGD, lbfgs, and adam, found the best way to identify DDoS attacks, which makes cybersecurity measures stronger". "And they also introduced voting and stacking classifiers, which combine predictions from more than one model", to show off

their creativity. This method makes predictions more accurate and makes them more resistant to a wider range of cyber-attacks. The combination of the SQLite flask for user and signin registration, along with the front-end used, is useful in real life. “Users may easily provide feedback, see forecasts, and engage with the system, which makes it more useful in the real world”.

6. FUTURE SCOPE

To make DDoS attack detection more accurate and efficient, the project may use future improvements in ML [21]. Looking into and using the latest algorithms and models may make the system even better at dealing with new cyber threats. The project's future goals include establishing ways to detect things in real time and plans to respond to them. To reduce the damage that DDoS assaults do to network resources, it will be important to combine technologies that make it easy to find and stop them quickly as they happen [23]. Using more complex behavioral analytic methods may help the project grow in the future. The system can better spot DDoS attack-related irregularities by learning how networks and devices normally behave. This allows proactive access to cyber security. As networks and cyber threats change, the future project objectives include ensuring that it can "grow and change with them". To make the system well-functioning in the constantly changing world of cyber security, "he will have to be able to control more data sets, various types of attacks and new technologies".

REFERENCES

[1] B. B. Gupta, P. Chaudhary, X. Chang, and N. Nedjah, ‘‘Smart defense against distributed denial of service attack in IoT networks using supervised learning classifiers,’’ Comput. Electr. Eng., vol. 98, Mar. 2022, Art. no. 107726.

[2] H. Beitollahi and G. Deconinck, ‘‘An overlay protection layer against denial-of-service attacks,’’ in Proc. IEEE Int. Symp. Parallel Distrib. Process., Apr. 2008, pp. 1–8.

[3] O. Yoachimik, ‘‘DDoS attack trends for 2022 Q1,’’ Cloudflare, CA, USA, Tech. Rep., Apr. 2022.

[4] T. Shorey, D. Subbaiah, A. Goyal, A. Sakxena, and A. K. Mishra, ‘‘Performance comparison and analysis of Slowloris, GoldenEye and Xerxes DDoS attack tools,’’ in Proc. Int. Conf. Adv. Comput., Commun. Informat. (ICACCI), Sep. 2018, pp. 318–322.

[5] L. F. Eliyan and R. Di Pietro, ‘‘DoS and DDoS attacks in software defined networks: A survey of existing solutions and research challenges,’’ Future Gener. Comput. Syst., vol. 122, pp. 149–171, Sep. 2021.

[6] J. Mirkovic and P. Reiher, ‘‘A taxonomy of DDoS attack and DDoS defense mechanisms,’’ ACM SIGCOMM Comput. Commun. Rev., vol. 34, no. 2, pp. 39–53, Apr. 2004.

[7] G. Somani, M. S. Gaur, D. Sanghi, M. Conti, and R. Buyya, ‘‘DDoS attacks in cloud computing: Issues, taxonomy, and future directions,’’ Comput. Commun., vol. 107, pp. 30–48, Jul. 2017.

[8] A. Bhardwaj, G. V. B. Subrahmanyam, V. Avasthi, H. Sastry, and S. Goundar, ‘‘DDoS attacks, new DDoS

taxonomy and mitigation solutions—A survey,'' in Proc. Int. Conf. Signal Process., Commun., Power Embedded Syst. (SCOPES), Oct. 2016, pp. 793–798.

[9] M. Sauter, '''LOIC will tear us apart' the impact of tool design and media portrayals in the success of activist DDOS attacks,'' Amer. Behav. Scientist, vol. 57, no. 7, pp. 983–1007, 2013.

[10] M. Antonakakis, T. April, M. Bailey, M. Bernhard, E. Bursztein, J. Cochran, Z. Durumeric, J. A. Halderman, L. Invernizzi, M. Kallitsis, and D. Kumar, ''Understanding the Mirai Botnet,'' in Proc. 26th USENIX Secur. Symp., 2017, pp. 1093–1110.

[11] B. Nagpal, P. Sharma, N. Chauhan, and A. Panesar, ''DDoS tools: Classification, analysis and comparison,'' in Proc. 2nd Int. Conf. Comput. Sustain. Global Develop. (INDIACom), Mar. 2015, pp. 342–346.

[12] P. J. Shinde and M. Chatterjee, ''A novel approach for classification and detection of DOS attacks,'' in Proc. Int. Conf. Smart City Emerg. Technol. (ICSCET), Jan. 2018, pp. 1–6.

[13] H. Beitollahi, D. M. Sharif, and M. Fazeli, ''Application layer DDoS attack detection using cuckoo search algorithm-trained radial basis function,'' IEEE Access, vol. 10, pp. 63844–63854, 2022.

[14] D. Kshirsagar and J. M. Shaikh, ''Intrusion detection using rule-based machine learning algorithms,'' in Proc. 5th Int. Conf. Comput., Commun., Control Autom. (ICCUBEA), Sep. 2019, pp. 1–4.

[15] W. Zhijun, L. Wenjing, L. Liang, and Y. Meng, ''Low-rate DoS attacks, detection, defense, and challenges: A survey,'' IEEE Access, vol. 8, pp. 43920–43943, 2020.

[16] Z. Wu, Q. Pan, M. Yue, and L. Liu, ''Sequence alignment detection of TCPtargeted synchronous low-rate DoS attacks,'' Comput. Netw., vol. 152, pp. 64–77, Apr. 2019.

[17] O. Boyar, M. E. Özen, and B. Metin, ''Detection of denial-of-service attacks with SNMP/RMON,'' in Proc. IEEE 22nd Int. Conf. Intell. Eng. Syst. (INES), Jun. 2018, pp. 000437–000440.

[18] R. SaiSindhuTheja and G. K. Shyam, ''An efficient metaheuristic algorithm based feature selection and recurrent neural network for DoS attack detection in cloud computing environment,'' Appl. Soft Comput., vol. 100, Mar. 2021, Art. no. 106997.

[19] S. Ramesh, C. Yaashuwanth, K. Prathibanandhi, A. R. Basha, and T. Jayasankar, ''An optimized deep neural network based DoS attack detection in wireless video sensor network,'' J. Ambient Intell. Hum. Comput., pp. 1–14, 2021.

[20] T. A. Tuan, H. V. Long, L. H. Son, R. Kumar, I. Priyadarshini, and N. T. K. Son, ''Performance evaluation of Botnet DDoS attack detection using machine learning,'' Evol. Intell., vol. 13, no. 2, pp. 283–294, Jun. 2020.

[21] P. Kumari and A. K. Jain, ''A comprehensive study of DDoS attacks over IoT network and their countermeasures,'' Comput. Secur., vol. 127, Apr. 2023, Art. no. 103096.

[22] S. Wankhede and D. Kshirsagar, ‘‘DoS attack detection using machine learning and neural network,’’ in Proc. 4th Int. Conf. Comput. Commun. Control Autom. (ICCUBEA), Aug. 2018, pp. 1–5.

[23] Y. Jia, F. Zhong, A. Alrawais, B. Gong, and X. Cheng, ‘‘FlowGuard: An intelligent edge defense mechanism against IoT DDoS attacks,’’ IEEE Internet Things J., vol. 7, no. 10, pp. 9552–9562, Oct. 2020.

[24] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, ‘‘Toward generating a new intrusion detection dataset and intrusion traffic characterization,’’ in Proc. 4th Int. Conf. Inf. Syst. Secur. Privacy, vol. 1, Jan. 2018, pp. 108–116.

[25] F. Ridzuan and W. M. N. Wan Zainon, ‘‘A review on data cleansing methods for big data,’’ Proc. Comput. Sci., vol. 161, pp. 731–738, Jan. 2019.

[26] J. Cai, J. Luo, S. Wang, and S. Yang, ‘‘Feature selection in machine learning: A new perspective,’’ Neurocomputing, vol. 300, pp. 70–79, Jul. 2018.

[27] M. Ahsan, M. Mahmud, P. Saha, K. Gupta, and Z. Siddique, ‘‘Effect of data scaling methods on machine learning algorithms and model performance,’’ Technologies, vol. 9, no. 3, p. 52, Jul. 2021.

[28] A. M. Mahfouz, D. Venugopal, and S. G. Shiva, ‘‘Comparative analysis of ML classifiers for network intrusion detection,’’ in Proc. 4th Int. Congr. Inf. Commun. Technol. Cham, Switzerland: Springer, 2020, pp. 193–207.

[29] M. Al-Zewairi, S. Almajali, and A. Awajan, ‘‘Experimental evaluation of a multi-layer feed-forward artificial neural network classifier for network intrusion detection system,’’ in Proc. Int. Conf. New Trends Comput. Sci. (ICTCS), Oct. 2017, pp. 167–172.

[30] D. Hunter, H. Yu, M. S. Pukish, III, J. Kolbusz, and B. M. Wilamowski, ‘‘Selection of proper neural network sizes and architectures—A comparative study,’’ IEEE Trans. Ind. Informat., vol. 8, no. 2, pp. 228–240, May 2012.

[31] M. J. Madić and M. R. Radovanović, ‘‘Optimal selection of ANN training and architectural parameters using Taguchi method: A case study,’’ FME Trans., vol. 39, no. 2, pp. 79–86, 2011.

[32] G. Panchal, A. Ganatra, Y. P. Kosta, and D. Panchal, ‘‘Behaviour analysis of multilayer perceptronswith multiple hidden neurons and hidden layers,’’ Int. J. Comput. Theory Eng., pp. 332–337, 2011.

[33] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, ‘‘Scikit-learn: Machine learning in Python,’’ J. Mach. Learn. Res., vol. 12, no. 10, pp. 2825–2830, Jul. 2017.

[34] L. Bottou, ‘‘Stochastic gradient learning in neural networks,’’ Proc. NeuroNimes, vol. 91, no. 8, p. 12, Nov. 1991.

[35] R. Wu, H. Huang, X. Qian, and T. Huang, ‘‘A L-BFGS based learning algorithm for complex-valued feedforward neural networks,’’ Neural Process. Lett., vol. 47, no. 3, pp. 1271–1284, Jun. 2018.

[36] A. S. Berahas and M. Takác, ‘‘A robust multi-batch L-BFGS method for machine learning,’’ Optim. Methods Softw., vol. 35, no. 1, pp. 191–219, Jan. 2020.

[37] I. K. M. Jais, A. R. Ismail, and S. Q. Nisa, ‘‘Adam optimization algorithm for wide and deep neural network,’’ Knowl. Eng. Data Sci., vol. 2, no. 1, pp. 41–46, 2019.

[38] D. P. Kingma and J. Ba, ‘‘Adam: A method for stochastic optimization,’’ 2014, arXiv:1412.6980.