IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

A Comparative Analysis Of DDOS Attack Detection Using Machine Learning And Deep Learning Algorithms

ISHITA KOLLURU

STUDENT

GITAM (DEEMED TO BE UNIVERSITY)

Abstract

We currently live in a digital era, where many of the services and products that we take are based online. This means that many of the activities that we perform in our day-to-day life are heavily dependent on the services existing and being functional 24/7 throughout the day. It's going to be taken into account for many of the industries in the world that exist today such as the government, healthcare and even business-related structures. In recent times, there's been a growth in distributed denial of service attacks, taking place on these services online. This attack takes place when a specific service or network is attacked by a large number of requests, which it is unable to process, slowing it down or even shutting it down in a few cases. This can be a very big issue when we are dealing with services that need to be maintained constantly to provide help to people around the world. As these forms of attacks grow, many organizations have been trying to come up with effective solutions to fight and overcome this issue. Although their systems are in place to combat distributed denial of service attacks, they often detect the attack too late to stop it. This is why we would like to create an effective method which is able to efficiently ensure that these forms of attacks are properly detected and proper actions are taken to ensure that they are stopped in their tracks. They're currently many machine learning models in the world today and we would like to go over how all of these machine learning models are able to work in the detection of distributed denial of service attacks. In this study we would like to identify the most effective machine learning methods when it comes to their detection and also assess them properly with respect to one another. We hope that our study is able to help many people pick the right methods when dealing with these attacks in order to create their own distributed denial of service attack detection model.

1 Introduction

Distributed denial of service attacks has become a very common form of malicious attacks that take place on various networks and services in the modern world. Unlike a denial-of-service attack, which takes place from a single source these distributed attacks take place from multiple locations throughout the world. This causes them to be very inconsistent and hard to detect due to their multiple sources that they originate from. These attacks stemmed from the fact that the attacker is able to disrupt the normal traffic of the targeted network or service by overwhelming it with a large number of requests. There are many different ways that a distributed denial of service attack can take place however they all tend to follow the same steps when they are initiated. In recent times, there were three major attacks that could be notable as being the largest malicious attacks currently known. The DYN attack, which took place in 2016 disrupted major websites such as Twitter, Netflix, and Spotify by exploiting various different IOT devices in order to form one of the largest bonnets and attack these services to disrupt them. The second most common attack in recent times was the Amazon web service attack that took place in 2020 which was conducted by a record-breaking 2.3 Tbps distributed of service attack, which completely brought down all of Amazon's web services. This attack was able to demonstrate how massively these forms of attacks have evolved in scale. Before this in 2018 GitHub faced a similar distributor to our live service attack which rounded up to be around 1.35 Tbps From the attacker which completely disrupted the platform. When these attacks take place, they tend to often slow down the network or services while also bringing them down when they are completely exhausted of resources. This is why they are very dangerous and need to be stopped to ensure that important services are not affected or brought to a stop.

To further understand these attacks, let's look into how they form and the steps in which an attacker goes through in order to initiate them. The first step in any form of distributed denial of service attacks is to create a botnet which the attacker will use as the multiple sources for their attack. This is done when the attacker is able to either compromise various devices around the world, such as computers or IOT devices by exploring vulnerabilities or simply using malware to take control of them. All of the devices that are joined in order to create the botnet are known as bots or zombies, and they're used in order to form a network which is remotely controlled by the attacker. Now that the attacker has successfully been able to create their botnet, they can now start to identify the target of their attack. The target itself will mostly comprise a specific server or even network and they will begin researching it extensively to exploit its weaknesses. Once a attacker is able to understand the weakness of their target. They can now begin to launch an attack by controlling the botnet to send a large number of requests to the target itself. There are multiple ways in which the attacker can choose to launch their attack and they arrange from volumetric attacks, protocol attacks, and application layer attacks. Each of these forms of attacks, target a different part of the target network and exploit them in order to bring it to a stop. When we look into volumetric attacks, these tend to focus on sending an overwhelming volume of traffic in order to ensure that the network is incapable of processing legitimate requests. This is done when the attacker is able to send a flood of data packets, which are often amplified by the target with the use of DNS amplification. When this is done, the target is unable to respond to the large number of DNS replies and is overwhelmed. By doing so this approach of attacking tries to target the bandwidth of the target and is completely exhausted so that it is not able to function efficiently. The second type of attack is a protocol attack which targets serve itself or the various firewall resources that it has. A common approach for this is the SYN flood in which the attacker is able to send a large number of TCP SYN requests to the target, but refuses to complete the handshake so that the server is constantly waiting while also simultaneously consuming all of its resources. In such a manner, it is exploiting the firewall it has in order to establish new connections and by doing so it is completely overwhelmed to serve itself by ensuring that the target allocates all of its resources to fight off malicious requests. The third and final type of these attacks is an application layer attack, which are far harder to attack due to the way in which they take place. There are too common methods when we look into this for attack and they are HTTP flooding and database query overloading. When the attacker chooses to perform a HTTP flood, they send the server a large number of HTTP GET and POST requests to it. By doing so each number of these requests consumes a large number of resources from the target, which eventually leads it to crash. When we look into databases where overloading the attacker is able to send specific queries to the target which triggers a very resource heavy database query, which completely slows down the target and can even crash it when the query itself is too large. This form of attack tries to target the application layer by simply using application logic which already exists within it. Due to this all of the attacks that take place look like normal users who are making legitimate requests to the server or service itself. Out of all three forms of tax this would be the hardest one to detect due to it mimicking legitimate traffic. No matter which attacks the attacker uses, the impact which it causes is drastic as it can completely slow down the target's network and bring it to a stop. Once the target network service is stopped, they will have to go through many various steps in order to ensure that they can bring them back online. In the modern world industries can take several days to recover from these forms of attacks and recover from them. Some of the industries that are commonly affected by distributed denial of service attacks are banking and financial services, healthcare, e-commerce, media and entertainment, and even government and public services. All of these industries are an important part of the modern world, and we must ensure that they are able to effectively combat these malicious attacks to ensure that they are able to function 24/7. If the services they provide go down even for a few minutes, it can completely impact their business and the revenue they would've made on that day. When we look into more important industries, such as healthcare, if any of the services they used are brought down by such a malicious attack it could even lead to the depth of many patients in various hospitals around the world.

This is why we would like to study these malicious attacks and create our own methods in order to combat and detect them. We would like to look into multiple different machine learning models and assess their ability in the detection of these attacks. By doing so we would like to not only compare them with each other, but also analyze their advantages and disadvantages that they bring to the table. In this paper we would like to explain thoroughly how our research was conducted and the various different models that we used in the later stages of our methodology. Before we could conduct our own research and begin creating our own classification models for the detection of distributed denial of service attacks we first had to look into existing methodologies and assess them. In the following section of paper, we would like to thoroughly analyze various studies that have been conducted in recent times to combat these attacks and look into the various challenges that they faced when doing so. We would also like to assess their strengths and weaknesses when dealing with various different forms of these attacks. Let us now take a look into the other studies that we were able to thoroughly look into.

2 Related Works

When you are able to study, various different research is conducted in the development of distributed denial of service attack detection methods. We focused on mainly looking into machine learning based methodologies. Most of the studies we looked into focused on creating these detection algorithms using supervised learning methods. This gave them various advantages by providing high accuracy and effectiveness for knowing the attacks. The major challenge that most of these models faced was acquiring the label data as it was very scarce on the Internet. Another major issue they stumbled upon was the vast amount of pre-processing that each one of these models needed in order to ensure that the data would fit the model itself. While these models were able to effectively detect large scale, distributed denial of service attacks, they often lack accuracy when dealing with small and normal attacks. The second methodology that we saw mostly used was unsupervised learning approaches, which were able to create effective detection methods by identifying various anomalies within the network traffic by using labeled data. Most of them were focused upon using clustering algorithms such as K means and other anomaly detection methods in order to build their model. The advantage of this approach was that it did not require labeled data and was able to train and make suitable predictions with the limited amount of data that it had. However, the major challenge that many of them faced was the fact that the model would often tend to have a very high false positive rate due to the lack of labeled data for the model to train on. Another issue that the model faced was its inability to assess normal traffic with malicious anomalies and often tend to trigger, even when there were legitimate requests on the server. One of the most interesting approaches was ensemble learning models, which were able to combine multiple different machine learning models as one. This gave them a clear advantage due to the fact that they were able to combine the strengths of these various models in order to achieve very high accuracy, as well as stability. It was also immune to overfitting and was able to generalize all of the data quite well. Well, this approach was able to bring very good results. It was limited by the fact that it would be impossible to implement in the real world due to its very long prediction times. On top of this, it also needed very large competition power to create due to the fact that multiple models are being used when making predictions. The final approach that we were able to study was reinforcement learning approaches, which were able to constantly monitor the real time environment and optimize it with reward-based policies. The major advantages that we saw with this approach was its ability to adaptively evolve to the various attack patterns the attacker makes a bit. Furthermore, it was very suitable for dynamic environments where the traffic was not stable and constantly changed throughout the day. The major challenge of creating such a model was the fact that it required real time simulations as it heavily depended upon real world conditions in order to train and assess its predictions. This method had the highest complexity and was also one of the highest computationally demanding methodologies. It was also one of the hardest pathologies to interpret and understand when it came to understanding how it was making decisions and establishing the policies for its reward system. In the future, if we are able to make its complexity less, we believe this would be one of the most effective ways to fight distributed denial service attacks due to its ability to adapt to the environment as well as the attack itself. Through these various different studies, we were able to understand that one of their major issues was dealing with the false positive rates. Most of them were computationally simple. We also observed that many of the existing models required a large amount of competition of power in order to constantly monitor the network. Taking all into account we would not like to move forward into creating our own intrusion detection model, and ensuring that he’s able to overcome the challenges that the other studies may have faced.

3 Methodology

For the creation of a machine learning model, we had decided to first analyze the multiple classification models that exist today. For this we had decided to go with DNN, KNN, SVM, decision tree, naïve bayes, quadratic, discriminant analysis, SGD, logistic, regression, and XGboost for our classifiers. Building models for each of these classification algorithms we wanted to analyze their accuracy and then use the best ones in order to build a robust and scalable detection model for distributed denial of service attacks. Through this project, we had to go over four major steps in which we had to properly process the data, establish the various class, machine, learning models, train and evaluate each model's accuracy, and finally combine all of them in order to create a single compact model, which would effectively act as our detection model. Let us dive deeper into all of these steps to understand the creation of our intrusion detection model.

3.1 Data Pre-Processing

The first step in our project was to ensure that the data was properly processed so that it could fit into our model. The Datta said that would be used in our research was the DDOS attack SDN dataset from mendeley data. Once we imported the data set into our project, we were able to analyze all of the columns to check for any null values. All the null values that we discovered in the dataset were assessed and the row which persisted them was dropped so that we could work with a complete dataset. Once this was complete, we had to ensure that each column had unique values, and there were no repeating ones which could mess with the training of our model. Once we insured all of the data was in a proper format to be fit into the model. We then began to analyze all of the data within the data. We did this by plotting the various functions that existed and visualizing the distribution of all of the categorical features. By doing so this helped us get an understanding of how the data set was distributed and all the important features within it. The final step in our pre-processing stage was to split the data into the training, set and testing set so that it could be used appropriately Once this was complete, we were now prepared to move onto the next part of our research where we worked on establishing the various different machine learning models.

3.2 Creation of Our Models and Their Analysis

As we have mentioned before, there were a total of 9 different machine learning models that we had decided to take and test in the creation of our intrusion detection system. The 9 models that we would be working with in our research were the DNN, KNN, SVM, decision tree, naïve bayes, quadratic, discriminant analysis, SGD, logistic, regression, and XGboost machine learning models. For each of these models, we first had to ensure that an individual model was created for all of them. Once we were able to efficiently define each model with its required layers, we were able to then work towards compiling the model to ensure that it existed in our research. At this stage, we had a total of nine different models which we had compiled and prepared to be fit with our dataset. Now that the models are prepared, we can now fit them with the training set and begin to evaluate them based on the accuracy they provide. Once all nine models had been properly trained and processed, we could now begin to plot the charts of the accuracy that they provided to get an understanding of the best models that we could choose from. The results for this will be displayed in the next section where all our results will be displayed together.

3.3 Hyperparameter Tuning & Selection for Our Model

From the various machine learning models that we had created, we were able to extract all of their hyper parameters to get an understanding of their training process. Once we have access to set up, we begin to collect all of these hyper parameters and tune them according to the results that they provided. Once this was done, we were able to select the best hyper parameters from the sponge and then use them in order to create our final model. For the creation of this model, we had to understand the best value of Epoch by running a total of 100 epochs and plotting graphs to understand where we were able to achieve the best results. In the stage we were able to identify that the best Apple value was 78 so we stuck with it. Now that we were able to create the final model all that was left was to fit it with the training set that we had acquired and allowed the model to train itself using it.

3.4 Testing and Evaluation of Our Model

Now that our model had been trained, all that was left was to assess it using the testing data set. It is important to assess the model using the testing data set as the model will be working with values that it has never seen before. This one sure that all the predictions and classifications it makes are 100% accurate and it was not able to remember the results from memory. We also had to ensure that during the evaluation of our model we were not encountering any false positive or true positive as this would also skew the true accuracy of the on itself. To do this, we used an ROC – AUC curve to ensure that this was not affecting the model that we had created. Overall, the hyper parameter model that we had created was able to achieve a detection rate of 99% with a very negligible false positive rate.

4 Results

The results that we have obtained throughout this project, I've been displayed below constituting of the accuracy of the various models that we have created along with the accuracy that we were able to generate with their final hyper parameter model.

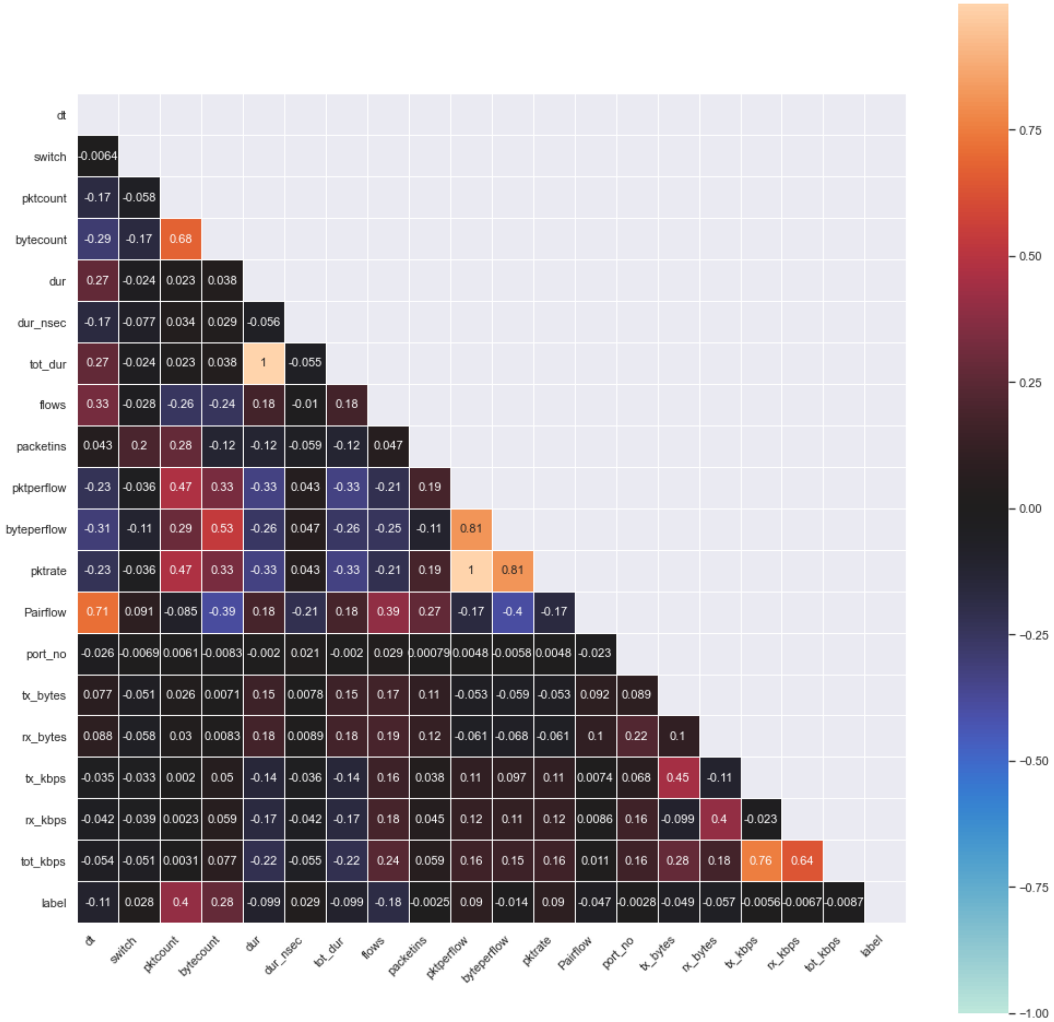

Figure 1: Heat map of Correlation of Features

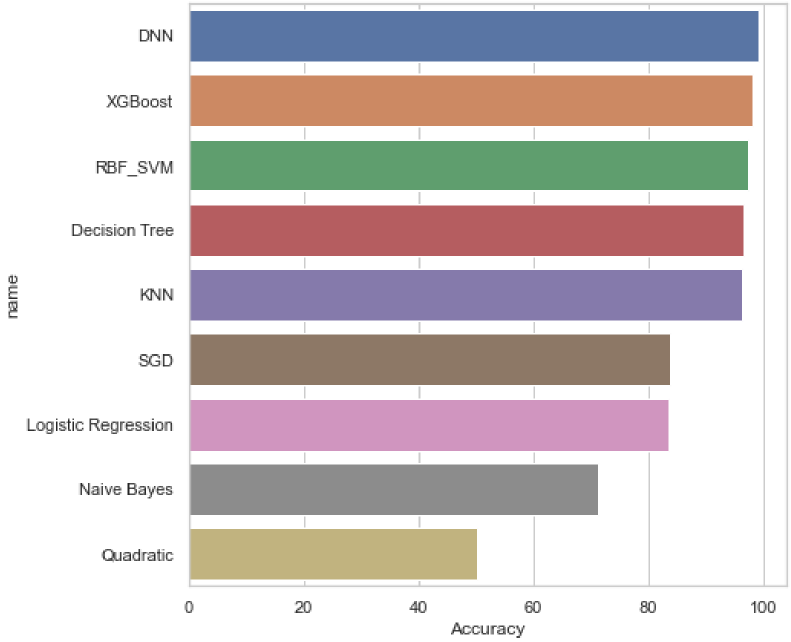

Figure 2: Comparison of Machine Learning Models

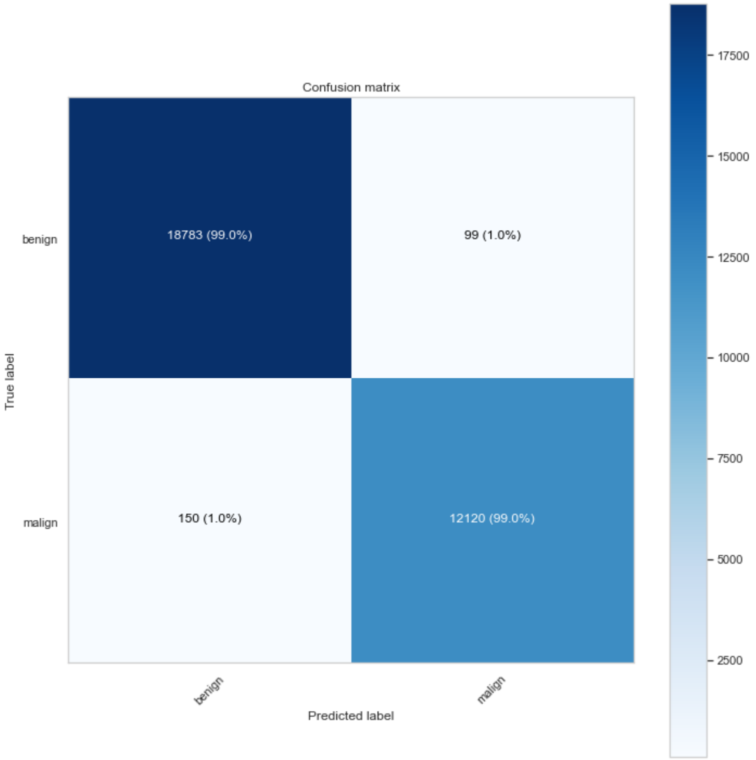

Figure 3: Confusion Matrix for Hyperparameter Tuned Model

5 Conclusion

Through this research, we were able to learn a lot about distributed denial of service attacks. Not only were we able to look into how these attacks took place, but we also came to realize just how malicious they are. Not only do they have the ability to completely slow down and limit the target server or network, but they are also able to bring it to a complete halt. This opened up her eyes to just how malicious various online attacks can be and how important it is for us to combat them in the proper format. We were able to analyze multiple different machine learning models, and look into how accurately they were able to detect these attacks. Through the study we were able to realize that deep neural networks were able to outperform all of the other machine learning models as they were able to gain an accuracy of 99% this was followed by XG boost and SVM ranking as the second and third best model models for the detection of this attack. Overall, we were able to use all the results that we compiled from the multiple models in order to use their hyper parameters to tune, and build our own model, taking the benefits of them all. This was able to give us a model that was able to accurately predict distributed denial of service attacks with a very low true positive rate. We believe this is a great achievement as we were able to accomplish this by using basic machine learning models and nothing complex. This greatly reduces the amount of computation of power when required in order to build an intrusion detection system. We have seen just how greatly these forms of attacks have grown in recent years and we have also analyzed just how malicious they can be. We hope that in the future, we are able to create more computational efficient systems which you're able to function just as efficiently as the one we created in our research. We hope that our findings will help many people in the creation of their own intrusion detection models with the thorough analysis of the various different machine learning models which we had used. We would like to extend our research by testing our model in real world scenarios by implementing it in various industries that we have access to. Before this is done, we would also like to consider testing it in our own sandbox environments to assess just how feasible it would be in the real-world scenarios. Overall, we believe we were able to learn a great deal through this research, not only on the topic of distributed denial of service attacks, but also with the creation of multiple machine learning models in order to assess them.

Acknowledgments We thank --- for a discussion on the framing of the project.

References

Wei Wang and S. Gombault, "Efficient detection of DDoS attacks with important attributes," *2008 Third International Conference on Risks and Security of Internet and Systems*, Tozeur, 2008, pp. 61-67, doi: 10.1109/CRISIS.2008.4757464.

A. Ramzy Shaaban, E. Abdelwaness and M. Hussein, "TCP and HTTP Flood DDOS Attack Analysis and Detection for space ground Network," *2019 IEEE International Conference on Vehicular Electronics and Safety (ICVES)*, Cairo, Egypt, 2019, pp. 1-6, doi: 10.1109/ICVES.2019.8906302.

T. Subbulakshmi, K. BalaKrishnan, S. M. Shalinie, D. AnandKumar, V. GanapathiSubramanian and K. Kannathal, "Detection of DDoS attacks using Enhanced Support Vector Machines with real time generated dataset," *2011 Third International Conference on Advanced Computing*, Chennai, India, 2011, pp. 17-22, doi: 10.1109/ICoAC.2011.6165212.

J. E. Varghese and B. Muniyal, "Trend in SDN Architecture for DDoS Detection- A Comparative Study," *2021 IEEE International Conference on Distributed Computing, VLSI, Electrical Circuits and Robotics (DISCOVER)*, Nitte, India, 2021, pp. 170-174, doi: 10.1109/DISCOVER52564.2021.9663589.

B. B. Gupta, A. Gaurav, V. Arya and K. T. Chui, "Efficient DDoS Attack Detection through Lightweight Deep Learning Model in Cloud Computing Environment," *2024 IEEE 24th International Symposium on Cluster, Cloud and Internet Computing Workshops (CCGridW)*, Philadelphia, PA, USA, 2024, pp. 208-212, doi: 10.1109/CCGridW63211.2024.00033.

Y. Huang, X. Fu, Q. Hou and Z. Yu, "The Early Detection of DDoS Based on the Persistent Increment Feature of the Traffic Volume," *22nd International Conference on Advanced Information Networking and Applications - Workshops (aina workshops 2008)*, Gino-wan, Japan, 2008, pp. 365-370, doi: 10.1109/WAINA.2008.160.

Y. Ling *et al*., "Real-time Detection of DDoS Attacks Based on Hurst Index," *2022 2nd International Conference on Networking Systems of AI (INSAI)*, Shanghai, China, 2022, pp. 42-45, doi: 10.1109/INSAI56792.2022.00018.

F. Reza, "DDoS-Net: Classifying DDoS Attacks in Wireless Sensor Networks with Hybrid Deep Learning," *2024 6th International Conference on Electrical Engineering and Information & Communication Technology (ICEEICT)*, Dhaka, Bangladesh, 2024, pp. 487-492, doi: 10.1109/ICEEICT62016.2024.10534545.

Y. -F. Hsu, A. Ryusei and M. Matsuoka, "Real Network DDoS Pattern Analysis and Detection," *2022 IEEE 46th Annual Computers, Software, and Applications Conference (COMPSAC)*, Los Alamitos, CA, USA, 2022, pp. 1489-1494, doi: 10.1109/COMPSAC54236.2022.00236.

S. Hameed and U. Ali, "Efficacy of Live DDoS Detection with Hadoop," *NOMS 2016 - 2016 IEEE/IFIP Network Operations and Management Symposium*, Istanbul, Turkey, 2016, pp. 488-494, doi: 10.1109/NOMS.2016.7502848.

R. S. Tambe, H. Dand and M. D. Salunke, "Role of Machine Learning Ensemble in DDoS Intrusion Detection," *2023 IEEE 5th International Conference on Cybernetics, Cognition and Machine Learning Applications (ICCCMLA)*, Hamburg, Germany, 2023, pp. 145-149, doi: 10.1109/ICCCMLA58983.2023.10346867.

Y. Chen, X. Ma and X. Wu, "DDoS Detection Algorithm Based on Preprocessing Network Traffic Predicted Method and Chaos Theory," in *IEEE Communications Letters*, vol. 17, no. 5, pp. 1052-1054, May 2013, doi: 10.1109/LCOMM.2013.031913.130066.

S. Yeom and K. Kim, "Improving Performance of Collaborative Source-Side DDoS Attack Detection," *2020 21st Asia-Pacific Network Operations and Management Symposium (APNOMS)*, Daegu, Korea (South), 2020, pp. 239-242, doi: 10.23919/APNOMS50412.2020.9237014.

A. Marques da Silva Cardoso, R. Fernandes Lopes, A. Soares Teles and F. Benedito Veras Magalhães, "Poster Abstract: Real-Time DDoS Detection Based on Complex Event Processing for IoT," *2018 IEEE/ACM Third International Conference on Internet-of-Things Design and Implementation (IoTDI)*, Orlando, FL, USA, 2018, pp. 273-274, doi: 10.1109/IoTDI.2018.00036.

J. -E. Chang, Y. -C. Chiu, Y. -W. Ma, Z. -X. Li and C. -L. Shao, "Packet Continuity DDoS Attack Detection for Open Fronthaul in ORAN System," *NOMS 2024-2024 IEEE Network Operations and Management Symposium*, Seoul, Korea, Republic of, 2024, pp. 1-5, doi: 10.1109/NOMS59830.2024.10575799.

X. Liang and T. Znati, "An empirical study of intelligent approaches to DDoS detection in large scale networks," *2019 International Conference on Computing, Networking and Communications (ICNC)*, Honolulu, HI, USA, 2019, pp. 821-827, doi: 10.1109/ICCNC.2019.8685519.

Z. Zhou, D. Xie and W. Xiong, "A P2P-Based Distributed Detection Scheme against DDoS Attack," *2009 First International Workshop on Education Technology and Computer Science*, Wuhan, China, 2009, pp. 304-309, doi: 10.1109/ETCS.2009.329.

M. Hassan, K. Metwally and M. A. Elshafey, "ZF-DDOS: An Enhanced Statistical-Based DDoS Detection Approach using Integrated Z-Score and Fast-Entropy Measures," *2024 6th International Conference on Computing and Informatics (ICCI)*, New Cairo - Cairo, Egypt, 2024, pp. 145-152, doi: 10.1109/ICCI61671.2024.10485097.

Z. Xu, Z. Yang, B. Di and L. Song, "Multi-Dimensional Security Indicator Design and Optimization for DDoS Detection in Edge Computing," *2023 IEEE 98th Vehicular Technology Conference (VTC2023-Fall)*, Hong Kong, Hong Kong, 2023, pp. 1-5, doi: 10.1109/VTC2023-Fall60731.2023.10333804.

I. Jemal, O. Cheikhrouhou and M. A. Haddar, "IoT DOS and DDOS Attacks Detection Using an Effective Convolutional Neural Network," *2023 International Conference on Cyberworlds (CW)*, Sousse, Tunisia, 2023, pp. 373-379, doi: 10.1109/CW58918.2023.00065.

Anuradha and A. Singhrova, "A host based intrusion detection system for DDoS attack in WLAN," *2011 2nd International Conference on Computer and Communication Technology (ICCCT-2011)*, Allahabad, India, 2011, pp. 433-438, doi: 10.1109/ICCCT.2011.6075142.

W. Jia, Y. Liu, Y. Liu and J. Wang, "Detection Mechanism Against DDoS Attacks based on Convolutional Neural Network in SINET," *2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, Chongqing, China, 2020, pp. 1144-1148, doi: 10.1109/ITNEC48623.2020.9084918.

S. Yadav and S. Subramanian, "Detection of Application Layer DDoS attack by feature learning using Stacked AutoEncoder," *2016 International Conference on Computational Techniques in Information and Communication Technologies (ICCTICT)*, New Delhi, India, 2016, pp. 361-366, doi: 10.1109/ICCTICT.2016.7514608.

L. Wang and Y. Liu, "A DDoS Attack Detection Method Based on Information Entropy and Deep Learning in SDN," *2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, Chongqing, China, 2020, pp. 1084-1088, doi: 10.1109/ITNEC48623.2020.9085007.

M. A. T. Laksono, Y. Purwanto and A. Novianty, "DDoS detection using CURE clustering algorithm with outlier removal clustering for handling outliers," *2015 International Conference on Control, Electronics, Renewable Energy and Communications (ICCEREC)*, Bandung, Indonesia, 2015, pp. 12-18, doi: 10.1109/ICCEREC.2015.7337029.