RESEARCH ARTICLE | JUNE 05 2024

An intelligent intruder framework for cyber-attacks using machine learning techniques FREE

Pushpalatha Sarla ✉; Bhavana Jamalpur; K. Chandhar

AIP Conf. Proc. 2971, 050004 (2024)
https://doi.org/10.1063/5.0196076

An Intelligent Intruder Framework for Cyber-attacks Using Machine Learning Techniques

Pushpalatha Sarla [1, a)], Bhavana Jamalpur[2], K. Chandhar[2]

[1]*Dept of Mathematics, Sumathi Reddy Institute of Technology for Women, Warangal, Telangana, India.*
[2]*School of Computer Science and Artificial Intelligence, SR University, Warangal, India.*

[a)] Corresponding author: pushpa.sarla@gmail.com

Abstract. Attacks like Distributed Denial of Service (DDoS) pose a major threat to the network's security. Many different firms' servers have fallen prey to such unusual types of attacks. These attacks from the many bots under the direction of the botmaster (cracker) can possibly result in the victim's computational and communication capabilities being severely impaired. To create an effective NIDS, the researchers used datasets that were made accessible to the public. Existing studies' datasets, however, are insufficient since they exclude the most commonly used protocols, such as DHCP, which is essential to network architecture. In a network, IP addresses and other crucial network setup settings are dynamically assigned via the Dynamic Host Configuration Protocol (DHCP). Two research inquiries serve as the foundation for this work: 1) what algorithm will get the best results for identifying Distributed Denial of Service attacks? 2) How accurate would these algorithms be if they were trained on real-world data? We exceeded 96% accuracy with the Random Forest Classifier, and we confirmed our findings using two measures. The results were also compared to other works to ensure that they were adequate. We also provide a thorough study to back up our conclusions.

Keywords: DDoSdetection, Machine Learning, security, network threats, dataset; DHCP; IP address

INTRODUCTION

Internet services are now crucial for governments, corporations, schools, and research. [1] This is because network-based services improve corporate flexibility, scalability, data accessibility, and maintenance simplicity [2]. The term "intrusion detection" refers to the practice of constantly keeping tabs on what's happening within a computer system or network, analysing the data for signals of attacks, and blocking those attempts wherever feasible. Generally, this is done by dynamically gathering data from a wide range of computer and network sources and then analysing the data for potential security issues. Packet filtering, authentication and authorization, and cryptographic protocols are all examples of traditional intrusion detection and prevention methods, but they can't prevent all threats, especially when attacks like denial of service get more complex. Furthermore, most frameworks based on such methods have poor misclassification detection performance and cannot continually adapt quickly to changing malicious behaviours.

Moreover, in the past several years, several ML approaches have been employed to address the issue of vulnerability scanning in an effort to increase detection capability and flexibility.Botnet-based DDoS attacks infect machines with malware to operate as a Botnet [3]. A DDoSattack chokes both computation and bandwidth, stopping the network. DDoSattacks have change to so corrupt that offenders now request ransom. The attack on banks to extract money is the modern example of cybercrime development [4]. 'Wanttocry-ransomware' and the Equifax and Deloitte breaches are more examples [5].A denial-of-service (DoS) attack is a type of cyber-attack in which a host player interrupts a computer or other device's usual operation. DoS attacks a targeted computer with requests until

International Conference on Research in Sciences, Engineering, and Technology
AIP Conf. Proc. 2971, 050004-1–050004-8; https://doi.org/10.1063/5.0196076
Published by AIP Publishing. 978-0-7354-4866-7/$30.00

regular traffic can't be handled, denying service to more users. Single-computer DoS attacks are common.A DoS attack oversaturates a machine's capacity, denying new requests. DoS attack vectors are grouped together by similarities. DoS threats fall into two groups.

Buffer Overflow attacks

Memory-based attack Buffer overflow can devour all storage space, RAM, or CPU time. This exploit causes sluggishness, system crashes, and other denial-of-service symptoms.

Flood attacks

A malicious actor might cause a denial-of-service attack by flooding a server with packets. Most DoS flood attacks require more bandwidth than the target.DoS attacks have historically leveraged network, software, and hardware weaknesses. DDoSattacks are more disruptive and easier to create, so they're less common. Most DoSattacks may become DDoS attacks

This type of attacks includes

a. Smurf attack—a previously exploited DoS attack in which a hostile actor uses a susceptible network's broadcast address to flood a targeted IP address.

b. **Ping flood**: This basic DoS attack uses ICMP (ping) packets to overwhelm a target. Overloading a target with pings A malicious actor might cause a denial-of-service attack by flooding a server with packets. Most DoS flood attacks require more bandwidth than the target.

c. A Ping of Death: This attack sends a faulty packet to a targeted workstation, causing system failures.

A distributed denial of service (DDoS) attack is a simultaneous DoS attack that originates from several sources. Usually, hundreds of thousands or even a million zombie machines are used to create a DDoS attack. These kinds of attacks involve "botnets" of computers that have already been infected with malicious software, allowing the attacker to remotely control them. A study says tens of millions of computers may be globally infected with botnet malware. DoS attacks are used by cybercriminals to demand money from businesses whose websites are essential to their operations. However, there have also been instances of reputable companies paying unscrupulous Internet users to harm their competitors' websites. Cybercriminals also use phishing and DoS attacks together to target users of online banks. They launch a DoS attack to take down the bank's website, and after that, they send phishing emails to clients to divert them to a phoney emergency page. DoS attacks are dominating the Internet and have proven to be quite profitable. According to the Network Infrastructure Security Report, DDoS attacks have increased in number since 2005. The largest attack in 2010 was twice as large as the largest attack in 2009, with one attack specifically hammering its target at 100 gigabits per second, as shown in Fig. 1.

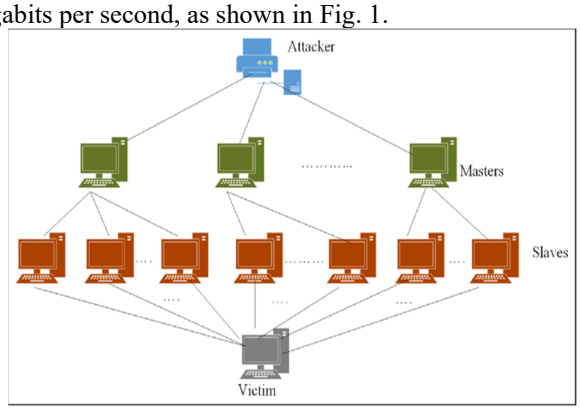


FIGURE 1. Structure of a DDoS attack.

LITERATURE REVIEW

In [11], the authors used 200K samples and 84 features from CICIDS2017 to evaluate features and make models. Feature engineering includes a correlation analysis and a tree-based feature significance investigation. Next, decision tree and SVM models were trained to classify DDoS and benign attacks. "Flow ID," "SYN Flag Cnt," and

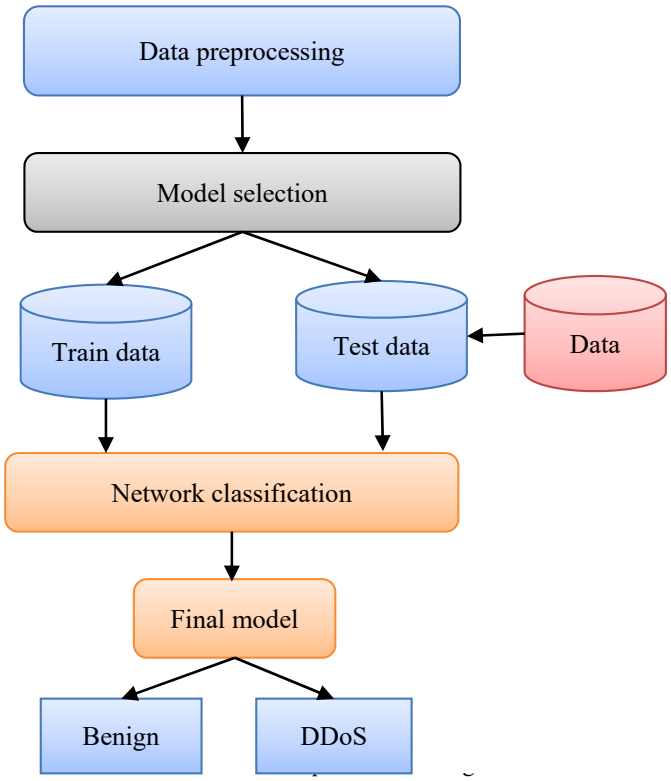
"Dst IP" affected attack detection the most. Machine learning algorithms identify DDoS attacks with near-perfect accuracy. Decision tree models outperformed SVMs. The results of this work replicated the original papers. According to the author [12], Weka experimenter's results with 10 classifiers on the KDD 20% training dataset show that random forest performs best with percent-correct, F-measure, and ROC (area underneath ROC). The Simplecart classifier comes in second in percent-correct and measure. The Simplecart classifier has the highest comparison field accuracy. ZeroR is the weakest classifier except for recall. With the dataset used for the experiment, further investigation might be limited to Random Forest, simple cart, J48, bagging, and IBk. This may reduce processing time and boost KDDCup20% data set classification accuracy. The author [13] investigates machine learning-based strategies to improve packet connection transfer predictions.

AI-based techniques might predict DOS, R2L, U2R, Probe, and big attacks. Results demonstrated that the suggested AI calculation technique may be compared with accuracy, recall, and F1 Score. This article [14] summarises the latest approaches to building IDS for cyber security. IDS is a good solution for cyber threats. Machine learning-based IDSs are accurate in a dynamic environment. This study explores which ML approaches have low accuracy to help researchers. This piece of work [15] We got 96% accuracy using the Random Forest Classifier and confirmed our results with two measures. It was compared to other work to ensure its accuracy. We also offer a full analysis of our findings. The author [16] discusses using machine learning to forecast malware attacks and developing a classifier to automatically identify and categorise events as "Has Detection or No Detection". Predicting malware penetration and network manipulation for cyber threat intelligence. To show work's usefulness, they employ a decision tree (DT) approach to evaluate a dataset. The dataset originated from Microsoft's Kaggle site. We identify smart grid hacks and utilise attack scenarios to assess penetrations and manipulations. The results suggest that ML could be used to find cyberattacks and predict future trends in cyber supply chains for smart grids.

METHODOLOGY

This study aims to construct an attack detection system with low false-positives and high true-positives. Decision Trees, Naive Bayes, Random Forest, and MLP (Neural Networks) were chosen because they perform differently and produce various results as shown in Figure 2.

Experiments use an emulated testbed topology. Wireshark captures network traffic. DHCP-specific data is created from Wireshark data. The experiment tests a server-side DHCP attack. DHCP Denial of Service and Starvation attacks were performed. The network has a DHCP server with 255 IP addresses. This DHCP server is attached to a central network switch that relays to five switches. Each network switch employs port mirroring to send network traffic to the core switch, where the IDS and Wireshark are attached. In this work we consider that each network switch includes 100 host machines and one attacker. All five networks have subnets of 10.0.0.0/24, 10.0.1.0/24, 10.0.2.0/24, 10.0.3.0/24, and 10.0.4.0/24. The DHCP server gets each network's initial IP address. Four environments were mimicked for each experiment. DHCP spoofing was followed by a DHCP starvation attack with a fluctuating 'chaddr' value but the same MAC address at layer 2. A server-side DHCP attack followed. Then, experiment-specific features were combined.



RESULTS AND DISCUSSION

On a machine running Windows 10 Professional, our test configuration ran Ubuntu as a virtual operating system as a virtual machine. in a laptop with an Intel Core 2 Duo CPU, 4 GB of RAM, Numerous tools were employed by the researchers. A Python-based DDoSD (DDoS detector) was utilised to create a Python-based DDoSD (DDoS detector) that produced five network classification models. It was done using the Intrusion Detection Evaluation Dataset [18]. This organisation has provided benchmark datasets for network security research since 1998. The dataset was made available. Unlike previous datasets used to identify network dangers, this one is contemporary. To provide a more "real" environment, this dataset includes recent attacks as well as legitimate traffic.

This dataset contains risks from the 2016 McAfee report related to the Web, brute force, DoS, DDoS, infiltration, Heart-bleed, bots, and scans. To train and evaluate our machine learning models for network classification, we used GBs of labelled data. Prediction accuracy is enhanced with a "good dataset." We looked at a lot of datasets before deciding on CIC IDS 2017. As more recent DDoSattacks have surfaced, other datasets like KDD99 are out of date and no longer relevant. We required a dataset that accurately represented the present. The ideal choice was CIC IDS 2017. 2,25,725 instances of 85 attributes make up our dataset. The workcompared intrusion datasets with other intrusion detection datasets [19]. Attack types, anonymity, protocols that are available, full capture, full interaction, full network configuration, full traffic, and feature set. Their research revealed that CICIDS2017 was the most recent and best of its kind compared to CAIDA, KDD99, DEFCON, ADFA, LBNL, KYOTO, etc.

Performance Evaluation metrics

Some metrics were calculated to assess the model's performance. These metrics aided in the model analysis and reflected the quality of the detection of the attacks for the specific machine learning algorithm used. The mentioned metrics are defined as follows:

1. Accuracy - Rate of data instances correctly classified by a model.

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Precision - Rate of correctly classified attacks divided by total of predicted attacks.

$$precision = \frac{TP}{TP + FP}$$

3. Recall - Rate of correctly classified attacks divided by the total number of attacks.

$$recall = \frac{TP}{TP + FN}$$

4. F1-score - is a measure of a model's accuracy on a dataset. [21] It is the harmonic mean of the precision and recall. [22]

$$F1 - score = \frac{2 \times recall \times precision}{recall + precision}$$

It was implemented. Each machine learning model was evaluated using different dataset component sizes, resulting in several train-test splits. Analyzed findings were tabulated. Tables 1 shows the results:

TABLE 1. Accuracy results collected for each algorithm

Train-Test	GNB	RF	DT	MLP
80-20	68.3780	88.8501	88.8400	87.8277
70-30	68.3670	88.8551	88.8501	87.7858
60-40	68.3890	88.8536	88.8502	87.9874
50-50	68.3964	88.8537	88.8503	87.6987
40-60	68.3543	88.8561	88.8504	87.3258
Average	68.3769	88.8537	88.8482	87.7210

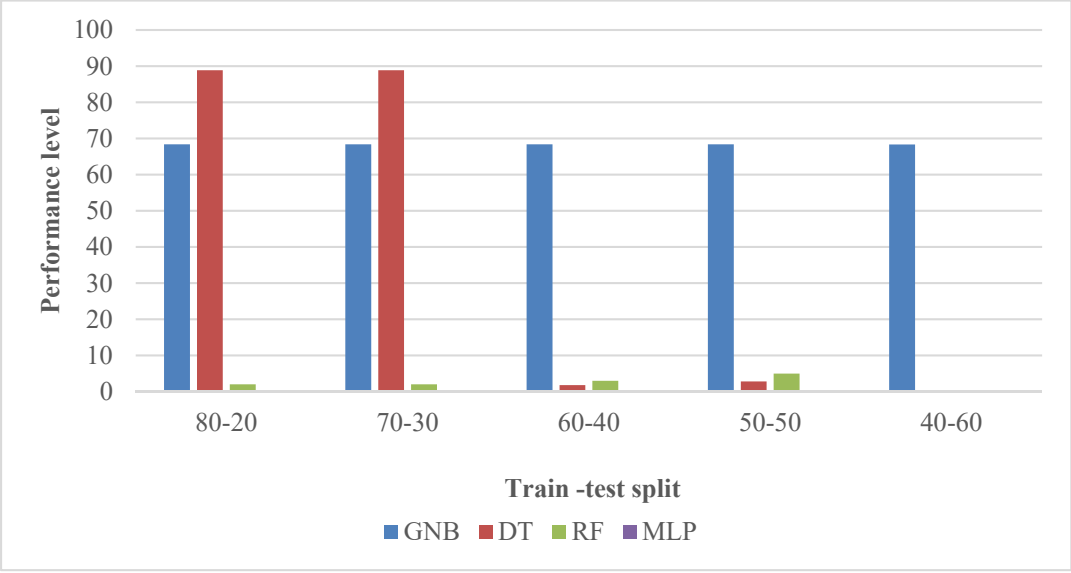


FIGURE 3. Performance comparison of four ML models

As can be seen in Fig. 3, for each assessment measure above, all models except the Gaussian Naive Bayes model. In order to compare the two models, we will average all of the dataset split values for each indicator. [23] The Gaussian Naive Bayes model is shown by the blue bars, the decision tree is represented by the orange bars, the random forest is represented by the green bars, and the MLP classifier is represented by the red bars. As a result, it is evident that Gaussian Naive Bayes was the model that fared the worst in comparison to the others (blue). The scores for accuracy and precision were 68.37 and 88.85, respectively. 87.72 for F1-score and 88.84 for Recall. For each parameter, the results were noticeably worse than the rest. Since this model is easier to understand than others, it runs faster and uses less computing power, so the result is about what was expected.

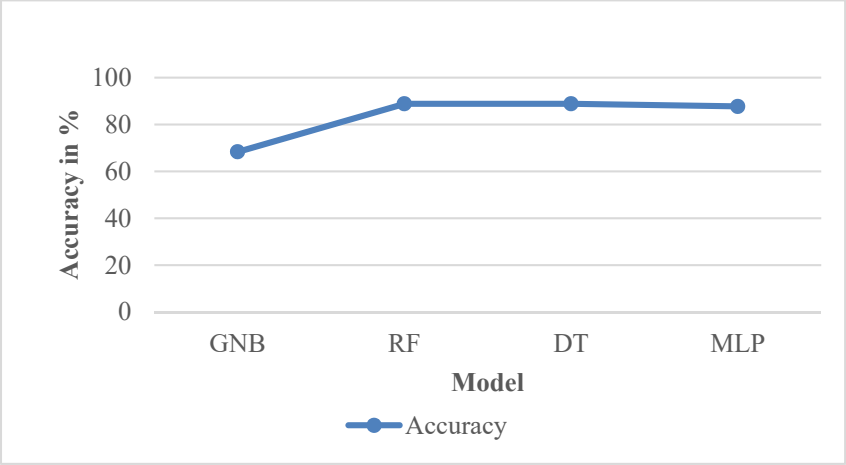


FIGURE 4. Performance comparison of Average Accuracy for four ML models

From Figure 4 it is clear that, handling vast volumes of data is not the best use for it. For example, data normalisation may be done to enhance this classifier. On the other hand, Random Forest (green), which is somewhat superior to Decision Tree, was the algorithm that performed better. The accuracy, precision, recall, and F1-score scores for the Random Forest classifier were 88.8400, 88.8501, 88.8502, and 88.8503, respectively. The scores were quite high, which indicates that the strategy was successfully implemented and the programme successfully recognised DDoSattacks. The decision trees functioned incredibly well, as was already indicated. This is due to the fact that this classifier functions similarly to Random Forest and produces results that are consistent with expectations. The MLP Classifier, which employs neural networks, was the model that required more time to do the training and didn't get the best outcomes. Neural network hyperparameter tuning can be used in future work to enhance the MLP classifier. This can identify and address various issues with neural networks. Everything that was previously appointed is reflected in the confusion matrices. It is obvious that there are significantly more incorrect predictions for the Gaussian Naive Bayes classifier, supporting the previously stated hypothesis. The same is true for Random Forest, which made fewer incorrect predictions.

CONCLUSION

The threat of a distributed denial of service (DDoS) attack poses a threat to network security. It is a hazard that will keep growing over time for apps and enterprises. Therefore, it is up to these platforms and corporations to make sure they are secure and to continue avoiding and identifying this sort of attack. In this paper, machine learning was employed to develop a high-quality DDoS detection system. It may claim that this objective was successfully attained. Despite just choosing 7 of the dataset's 83 features, it had a large number of features. The Gaussian Naive Bayes, Random Forest, and MLP Classifier were the classifiers employed. Random Forest performed the best. The analysis using various train and test sample sizes revealed that it had no appreciable effect in this instance. This may have happened as a result of the quality of the model and well-chosen features in the model. The model will be put to the test in a real-world setting with live network traffic capture of various DDoS attacks, and efforts will be made to further enhance the MLP and GaussianNB outcomes.

REFERENCE

1. Comptia. What is a ddos attack and how does it work? https://www.comptia.org/content/guides/what-is-addos-attack-how-it-works.
2. Khamooshi, G. P. (2019). The benefits of using web-based applications. https://www.geeks.ltd.uk/insights/the-benefits-ofusing-web-based-applications.
3. Hacking Incidents, 2018. https://en.wikipedia.org/wiki/List_of_security_hacking_incidents. Accessed February 15, 2018.
4. Transformation of DDoS attacks in Global warefare, 2018. https://qz.com/860630/ddos-attacks-have-gonefrom-a-minor-nuisance-to-a-possible-new-form-of-global-warfare/. Accessed January 1, 2018.
5. T. Subbulakshmi et.al, " A Unified Approach for Detection and Prevention of DDoS Attacks Using Enhanced Support Vector Machine and Filtering Mechanisms", ICTACT Journal on CommunicationTechnology, June 2013.
6. M. Alkasassbeh, G. Al-Naymat et.al," Detecting Distributed Denial of Service Attacks Using Data Mining Technique," (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 7, pp. 436-445, 2016. Science and Information Technologies, Vol. 6 (2), pp. 1096-1099, 2015.
7. M.Aijaz, S. Parveen," Analysis of Dos and DDoS Attacks", International Journal of Emerging Research in Management &Technology, Volume-5, Issue- 5.2012.
8. NourMoustafa, Jill Slay, "Creating Novel features to Anomaly Network Detection using DARPA-2009d Data set", School of Engineering and Information Technology, Australia, July 2015.
9. Niharika Sharma, AmitMahajan, VibhakarMansotra, Identification and analysis of DoS attack Using Data Analysis tools," International Journal of Innovative Research in Computer and Communication Engineering, Vol. 4, Issue 6, June 2016.
10. Sarraf, Saman. (2020). Analysis and Detection of DDoS Attacks Using Machine Learning Techniques. American Scientific Research Journal for Engineering, Technology, and Sciences. 66. 95-104.
11. .N, Udayakumar&Lakhara, S.P. & T Subbulakshmi, Dr. (2017). Detection of attacks using machine learning techniques. International Journal of Economic Research. 14. 339-351.
12. S., Priyadharsini. (2021). PREDICTION OF NETWORK ATTACKS USING MACHINE LEARNING TECHNIQUES. International Journal of Engineering Applied Sciences and Technology. 5. 10.33564/IJEAST.2021.v05i10.017.
13. Tripathi, Vikas&Devesh, Pratap& Singh, Bhaskar& Pant, Vijay & Kumar, Vijay. (2019). A Comprehensive Analysis and Solution of Cyber Attacks using Machine Learning Techniques. International Journal of Innovative Technology and Exploring Engineering. 8. 70-74. 10.35940/ijitee.L1019.10812S319.
14. Bindra, Naveen &Sood, Manu. (2019). Detecting DDoS Attacks Using Machine Learning Techniques and Contemporary Intrusion Detection Dataset. Automatic Control and Computer Sciences. 53. 419-428. 10.3103/S0146411619050043.
15. Yeboah-Ofori, Abel. (2021). Classification of Malware Attacks Using Machine Learning In Decision Tree. 11. 10-25.
16. Mallikarjuna Reddy Doodipala, Pushpalatha Sarla**,** VamshikrishnaThandu "Markovian Model for Internet Roughter Employing PBS Mechanism Under Self-Similar Traffic", *AIP Conference Proceedings 2246,* 020064 (2020); https://doi.org/10.1063/5.0014437 ICMSA-064, GITAMUniversity,Hyderabad.
17. Sci-kit Learn, Machine Learning in Python, 2017. http://scikit-learn.org/stable/. Accessed November 5, 2017.
18. V. Pranathi, G. Ranadheer Reddy, G. Sunil, D Raghava Kumar, BhavanaJamalpur "A Comprehensive Study on the Various Applications of Deep Learning" *IOP Conference Series: Materials Science and Engineering,* vol.981, issue.2,2020, https://iopscience.iop.org/article/10.1088/1757-899X/981
19. Pushpalatha Sarla**,** D MallikarjunaReddy, Manohar D,Ravikiran G "Analytical study on Air India Traffic Using Artificial Neural Networks" *IOP Conf. Series*: Materials Science and Engineering 981 (2020) 022097. doi:10.1088/1757-899X/981/2/022097
20. Pushpalatha Sarla, S.Rakmaiah, Archana Reddy, "Forecasting the spread of Covid-19 pandemic outbreak in India using ARIMA time series modelling" *AIP Conference Proceedings* 2418, 060003 (2022); https://doi.org/10.1063/5.0081944, 24 May 2022.
21. A.M Girija, Mallikarjuna Reddy, Pushpalatha Sarla "Study on patients arrival at hospital using queuing model with self-similar characteristics" *AIP Conference Proceedings* 2418, 060005 (2022); https://doi.org/10.1063/5.0081946, 24 May 2022

22. Pushpalatha Sarla, A.M Girija, R Archana Reddy, "Analysis of patient arrivals for COVID-19 test at healthcare center in Vijayanagara District" *AIP Conference proceedings*,2418, 060002 (2022); https://doi.org/10.1063/5.0081939, 24 may 2022.
23. Pushpalatha Sarla**,** D. Mallikarjuna Reddy, Manohar Dingari, "Queue Length-Busy Time Distribution of Web Users Data with Self-Similar Behavior Proceedings of the *International Conference on Innovations and Advancements in Computing* –ICIAC, March 2016**,** https://doi.org/10.26438/ijcse/v7i4.427441 GITAM University, Hyderabad.
24. A.M Girija, Mallikarjuna Reddy**,** Pushpalatha Sarla, "Covid-19 Patient Arrival Pattern with self-similarity at Health care center using queuing models, *Advances & Applications in Mathematical Sciences*, ISBN 978-93-90146-22-2.