International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

DDOS Attack Detection Using Artificial Intelligence

Mrs. D. Bhavya Varma[1]*, Puppala Deepika*[2]*, Songa Om Poojitha*[3]

Assistant Professor, Dept. of Computer Science and Engineering(Cyber Security), Marri Laxman Reddy Institute of Technology and Management, Hyderabad
B.Tech students, Dept. of Computer Science and Engineering(Cyber Security), Marri Laxman Reddy Institute of Technology and Management, Hyderabad

ABSTRACT:

In response to the increasing threat of DDoS (Distributed Denial of Service) attacks, this project investigates fortifying defences against such malicious invasions. The project incorporates a user-friendly UI featuring two buttons: one for uploading captured traffic files and another for analysis to classify whether it's a DDoS attack. The background of the problem aspires to a robust and adaptive DDoS detection system to ensure the continuity of online services. To resolve this, the project proposes an automated DDoS attack detection mechanism powered by Machine Learning and Artificial Intelligence. The application involves two pivotal experiments: the first assesses model accuracy, highlighting the Decision Tree as the most promising, while the second focuses on preventing overfitting during training, and the Random Forest Classifier stands out to this one. The challenges encountered were mitigated through techniques like early stopping and regularization. The model's application across various scenarios showcased its potential for effective real-time DDoS detection.

Keywords: Traffic Analysis, DDoS attack, Cyber Security, Anomaly Detection, AI, Recognition, Detection

INTRODUCTION:

A DDoS(Distributed Denial of Service) attack is a malicious attempt to disturb the normal traffic of a targeted server or network by overwhelming the target or its surrounding infrastructure with a flood of Internet traffic. In cybersecurity, we think of the CIA triad in terms of types of attacks:
Confidentiality: Is my information secret?
Integrity: Is my information accurate and trustworthy?
Availability: Can I get my information when and where I need it?
Unlike other types of attacks, attackers do not use DDoS to breach your security perimeter. Instead, DDoS attacks primarily target the availability aspect of the CIA triad. DDoS attacks aim to exhaust a server or network with a massive volume of traffic, leading to a sudden surge in requests. Consequently, legitimate users are unable to access the targeted services, causing service downtime. The disruptive nature of DDoS attacks has the potential to inflict significant financial losses on businesses, especially those that heavily rely on uninterrupted online availability. Industries such as e-commerce websites and online services are particularly susceptible to these attacks, given their independence on continuous accessibility for sustaining operations. It makes them prime targets for malicious hackers seeking to exploit vulnerabilities in the digital world. As a result, organizations must invest in robust DDoS mitigation strategies to protect their online infrastructure.

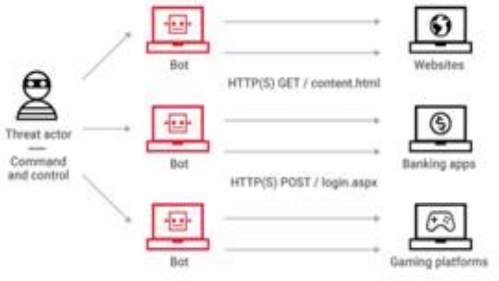


Fig 1 DDOS Attack

The initial methodology focuses on data preprocessing and traditional machine learning models to detect cybersecurity threats. It struggles with dimensionality reduction, therefore it potentially loses critical information. In response, our project enhances this model by incorporating a broader list of algorithms, and we focus on fine-tuning and expanding preprocessing measures as we aim for a more nuanced detection capability. The second approach leverages the flexibility of Software Defined Networking (SDN) to counteract DDoS threats. However, its reliance on conventional detection methods may lead to delays and false positives. Our enhancement involves advanced hyperparameter optimization and evaluating models with extensive metrics, which aim for prompt and accurate threat detection in a dynamic network environment. This final methodology employs multiple linear regression on a benchmark dataset for threat prediction, based on their assumption of linear relationships. Our project seeks to outperform this by exploring models that can analyze complex, nonlinear interactions. By doing so, it can deliver a robust and versatile detection system. Also, it transcends the linear confines of the previous approach. By harnessing Kaggle's DDoS datasets, we could take advantage of Machine Learning to produce a comprehensive DDoS detection model and deploy it to forestall any potential DDoS threat factor. The detector is responsible for monitoring incoming network traffic and responding to that traffic with the corresponding prediction for DDoS. If a server manager is hired to organize and recognize malicious requests instead of using Artificial Intelligence, the human errors of omitting can be devastating. On the other hand, hackers in other countries use the time difference to launch DDoS attacks on servers in the early hours of the morning, when people are already resting. At this time, the server is very vulnerable due to insufficient awareness. On the contrary, Artificial Intelligence can easily address human instinctive issues.

Additionally, the second effective method to fortify network security is to activate a Web Application Firewall (WAF). A Web Application Firewall serves as a proactive defense mechanism by "sanitizing" incoming traffic and filtering out malicious requests before they even reach the targeted server. By implementing a WAF, organizations can significantly reduce the risk of DDoS disruptions. WAF acts as a shield that deflects malicious traffic and allows traffic from legitimate users. However, they have limitations when compared to ML methods. Firewalls operate based on predefined static rules and signatures. It makes them effective at blocking known attack patterns but less adaptive to evolving attacking threats. They also generate more false positives, block legitimate traffic, and struggle to detect complicated and dynamic DDoS attacks. In two distinct experiments, we targeted to evaluate the accuracy metric of our 5 ML algorithms using the 20% test dataset and tried to mitigate the potential risk of overfitting. In the first experiment, we evaluated the performance of Logistic Regression, KNN, SVM, Decision Tree, and Random Forest. We trained and validated them with the same 80% of the dataset (using the same random state). The separated test dataset is used to assess their performance. Notably, the Decision Tree classifier consistently stood out with the best accuracy. Reasonably, its high accuracy could be attributed to its ability to adeptly capture complex patterns within the dataset. In the second part of the experiment, we shifted our attention to avoid overfitting, as several signs of this critical concern were displayed while evaluating the model. By implementing overfitting killer techniques such as cross-validation, regularization, ROC AUC, and early stopping, we aimed to decrease overfitting. As a result, the Random Forest classifier, with some hyperparameter tuning, delineated the highest reduction in overfitting, ultimately enhancing its ability to generalize effectively to unseen data and showcase increased accuracy when it is implemented in real-time. These meaningful experiments portrayed the strengths and weaknesses of various models and pointed out the importance of overfitting mitigation strategies.

OBJECTIVE:

The main goal of this project is to create an intelligent DDoS detection system using Artificial Intelligence (AI) and Machine Learning (ML) approaches to precisely identify and block Distributed Denial of Service (DDoS) attacks in real-time. With DDoS attacks getting increasingly sophisticated and massive, legacy detection techniques usually fail to deliver timely and effective countermeasures. The goal of this project is to develop a resilient and adaptive detection process that not only identifies incoming network traffic as benign or malicious but also improves the general security posture of online services. Through the use of sophisticated machine learning algorithms, the system hopes to enhance detection rates, minimize false positives, and promote availability of vital online resources.

Aside from improving detection functionality, the project also seeks to establish an easy-to-use interface for interaction with the detection system. Through the creation of a Tkinter application and a Flask web application, the project hopes to democratize access to advanced DDoS detection for network administrators and security specialists of diverse technical proficiency. The user interface will enable users to simply upload traffic information, trigger analysis, and obtain actionable knowledge about possible DDoS attacks. Finally, the project aims to enable organizations to actively protect themselves from DDoS attacks, limiting service downtime as well as protecting their digital estate.

In addition, the project plans to make an overall contribution to the field of cybersecurity by presenting an integrated methodology for DDoS detection that will be adjustable and extendable for different applications. By sharing the methodologies, issues, and solutions it has encountered during development, the project will be a useful reference for subsequent research and development in network security. This is to include investigating the incorporation of other machine learning models, improving data preprocessing methods, and adding real-time monitoring functionality. In sharing findings and insights, the project aims to promote collaboration and innovation within the cybersecurity community with a view to developing stronger defenses against the increasingly dynamic field of cyber threats.

SCOPE:

The project scope covers the design and development of a robust DDoS detection system that incorporates intelligent machine learning features to detect and counter DDoS attacks accurately. The project aims to implement a strong system that can be used to evaluate network traffic real-time, filtering out legitimate behavior from malicious attempts. By utilizing different machine learning algorithms, including Random Forest, Support Vector Machine (SVM), and K-Nearest Neighbors (KNN), the system will try to be highly accurate in detection while keeping false positives to a minimum. The project will also investigate the application of ensemble methods and feature engineering to further improve the performance of the model so that it can keep up with the changing nature of DDoS attacks. This emphasis on real-time analysis is important, as prompt detection and response are necessary to ensure the availability of online services and prevent potential financial and reputational loss.

Aside from the DDoS detection technicalities, the project shall also focus on user experience in that it shall create easy-to-use interfaces by way of a Tkinter app and a web application using Flask. These shall enable users to upload traffic files with ease, run analyses with a single command, and present them with crisp, actionable findings on possible DDoS risks. The scope includes designing the user interface to be accessible to individuals with varying levels of technical expertise, ensuring that network administrators and security professionals can effectively utilize the system without requiring extensive training. Furthermore, the project will incorporate features such as result logging and reporting, enabling users to maintain records of analyses for future reference and compliance purposes. By prioritizing usability, the project sets out to bridge the gap between intricate machine learning processes and real-world application in practical scenarios.

The project will also go beyond its original scope to encompass documentation and knowledge sharing within the cybersecurity community. This includes documenting the methodologies used, challenges encountered, and solutions implemented throughout the life cycle of the project. Through extensive documentation, the project will be a useful tool for researchers and practitioners who are interested in DDoS detection and prevention. The project will also investigate possible directions for future improvements, including incorporating other machine learning models, enhancing data preprocessing methods, and adding real-time monitoring features.

CHALLENGES:

1. Model Architecture

Training AI models for DDoS detection is naturally complicated by the complex architecture that tends to include millions of parameters. The parameters are necessary for the model to learn and generalize from the data but also impose tremendous computational requirements. The training process often demands powerful hardware, e.g., GPUs or TPUs, to deal with the vast calculations involved in the tuning of these parameters. As the model learns, it has to go through a high-dimensional space in which relationships between the features may be non-linear and subtle. Not only does such complexity raise the training time, but it also requires close supervision to guarantee that the model converges to a good solution without getting trapped into local minima. In addition, fine-tuning the hyperparameters of the model is an important step that could greatly impact its performance. Hyperparameters like learning rates, batch sizes, and the number of layers need to be carefully tuned to obtain optimal performance. This can be time-consuming and usually requires the execution of several experiments to determine the best configuration. Overfitting is also a common problem in model training, especially with deep architectures. When a model overlearns the training data, it could end up failing to generalize new data and may perform poorly when applied in the real world. To avoid overfitting, methods like regularization, dropout, and cross-validation are used but need to be implemented carefully and can make training even more cumbersome.

2. Dataset

The nature of the dataset used during the training of machine learning models matters a lot to obtain credible and accurate results. Raw data sets usually bear many errors and inconsistencies like missing values, duplications, and irrelevant attributes, which may harm the learning process of the model. Thorough data cleaning and preprocessing should be done prior to training to verify that the data set is high quality. Such a process could include the determination and correction of inaccuracies, the replacement of missing values, and the removal of any duplication that might affect the results. A properly prepared dataset does not only improve the performance of the model but also increases the reliability of predictions made during real-time DDoS detection. Aside from cleaning, data diversity in the dataset is another challenge. Various types of data like categorical, numerical, and text data call for separate preprocessing procedures. For instance, categorical data might be required to be encoded into numerical representations and numerical data could be normalized such that all features have an equal effect on the learning process of the model. In addition, unbalanced datasets are an issue of concern in DDoS detection because the instances of benign traffic are typically much larger than the instances of malicious traffic. This could result in skewed prediction such that the model prefers the majority class. Methods like oversampling, undersampling, or employing synthetic data generation techniques like SMOTE can mitigate this problem, but they also introduce complexity to the data preparation process.

3. Data Visualization

Good data visualization is a foundation of data analysis, allowing researchers and stakeholders to make sense of complex datasets and convey insights effectively. The selection of visualization type—bar graphs, pie charts, line graphs, or scatter plots—would be based on the type of data and the particular insights to be communicated. Each type of visualization has its advantages and disadvantages; for example, bar graphs are great for comparing discrete categories, whereas line graphs are better suited to depict trends over time. Choosing the right visualization needs a strong insight into both data and the narrative to be conveyed so that the visual depiction can truly represent the underlying data. Additionally, effective data visualization demands careful design components, such as color palettes, annotations, and legends. Inadequately designed visualizations may create incorrect interpretations of the data, hiding the key insights instead of revealing them. Hence, it is important to make sure that visualizations not only look good but are also informative and clear. Interactive visualizations can further contribute to engaging the user to a greater extent by enabling users to drill down, zoom, or filter particular features of the data. These can yield rich insights that might not be gained through static visualizations. But interactive visualizations need more technical skills and user experience considerations to make sure they convey the message as intended.

SOLUTIONS:

1. AI Components

The AI elements of the DDoS system are the central part of the solution, where machine learning is used to scan and categorize network traffic behaviors characteristic of a DDoS attack. Random Forest, SVM, and KNN are some of the leading models used in this task due to their exclusive advantages that favor them for use here. For instance, Random Forest is an ensemble learning method that combines multiple decision trees to improve accuracy and reduce the risk of overfitting. This model excels in handling high-dimensional data and can effectively capture complex interactions between features, making it particularly adept at distinguishing between benign and malicious traffic. In addition to model selection, the training process involves rigorous data preprocessing and feature engineering to enhance the models' performance. This involves normalizing numeric features, encoding categorical variables, and handling class imbalances using methods like SMOTE. The models are trained on a complete dataset with both malicious and benign traffic samples so that they can learn the discriminative features of DDoS attacks. Hyperparameter tuning is also an important part of model performance optimization since it entails the tuning of parameters like the number of trees in a Random Forest or the type of kernel in an SVM. Through systematic testing of various configurations, the models can be optimized to provide the optimal accuracy and generalization ability, thus providing effective real-time detection of DDoS attacks.

2. Tkinter App

The Tkinter application is a friendly graphical user interface that enables easy interaction with the DDoS detection system by the users. Having usability as its guiding principle, the application has a simple design featuring simple buttons for uploading traffic data files and starting the detection process. The ease with which users can find their way through the interface makes it easily usable even by users with minimal technical know-how. After a file is uploaded, the app analyzes the data through the pre-trained machine learning algorithms and gives users instant feedback on whether the traffic is a symptom of a DDoS attack. The analysis is given in a simple and easy-to-understand format so users can make the best possible decision based on it. Besides its primary functionality, the Tkinter app also contains logging and reporting features. Results of analyses can be saved by users to text files, which can be helpful for documentation and further research. The design of the app focuses on responsiveness and simplicity so that users can easily comprehend the output and take necessary action if a DDoS threat is identified. In addition, the app can be extended with other features, such as live monitoring and notifications, to further improve its functionality and offer users a complete tool for network security management. In general, the Tkinter application is instrumental in closing the gap between intricate machine learning routines and user interaction to bring sophisticated DDoS detection to more people.

3. Flask Web Application

The Flask web application is the backend part of the DDoS detection system that offers a secure framework for user request handling and data processing. Developed on the Flask microframework, the application enables users to access the detection system via a web interface with flexibility and accessibility. Customers can upload traffic data files or manually enter information via a web form, making it simple to initiate the detection process from anywhere with an internet connection. The Flask app handles such inputs and uses the pre-trained machine learning models to scan the traffic and make predictions about potential DDoS attacks. One of the main strengths of the Flask web application is the capability to complement numerous front-end technologies, adding a rich experience to the end user. With the use of Bootstrap for responsiveness, the application provides a responsive and visually striking interface that runs well on numerous devices and sizes of screens. The application further integrates user authentication capabilities, facilitating secure access into the DDoS detection platform. This is especially crucial in settings where sensitive information is being processed. The findings of the analysis are then reported back to the user in a straightforward manner, allowing them to efficiently evaluate the circumstances and take required actions. In general, the Flask web application supports the Tkinter app by offering a flexible and scalable platform for DDoS detection and addressing varied user requirements and preferences.

The user either opens the Tkinter application or the Flask web application to upload a traffic data file or manually enters the data for the web application only. The back end then starts the DDoS detection procedure after the UI program sends a request to the bank end, acting as a bridge between the AI components and the user interface. The trained AI models are used within the application to examine the uploaded traffic statistics. Next, based on these AI models' predictions, one can determine whether the traffic data shows signs of a DDoS attack. The application receives the generated prediction results and displays them to the user. Users can take appropriate action in response to the detected threat and make well-informed decisions thanks to this intuitive interface.

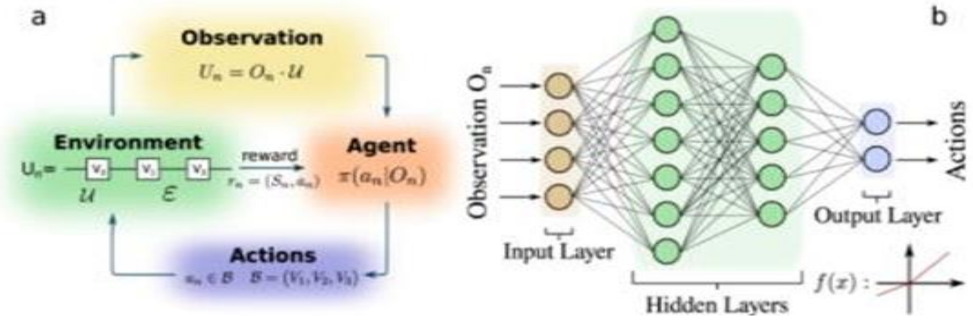


Fig 2 System Overview

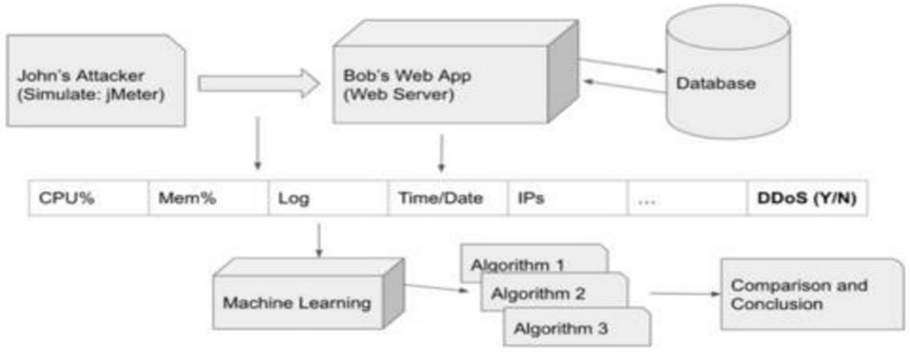


Fig 3 Overview of Solution

The key purpose of having an AI model is that it has the role of detecting abnormal traffic. We have tested out popular ML models such as KNN. KNN is an ML algorithm that assigns a label to a data point based on the majority class of its k closest (usually measured by Euclidean Distance) neighbors in the feature space, making it particularly useful for pattern recognition and similarity-based predictions. Its ability to classify instances based on their similarity to all data points makes it very suitable for a DDoS detection system. However, it follows proper hyperparameter tuning and determination of the value K for achieving highly accurate results.

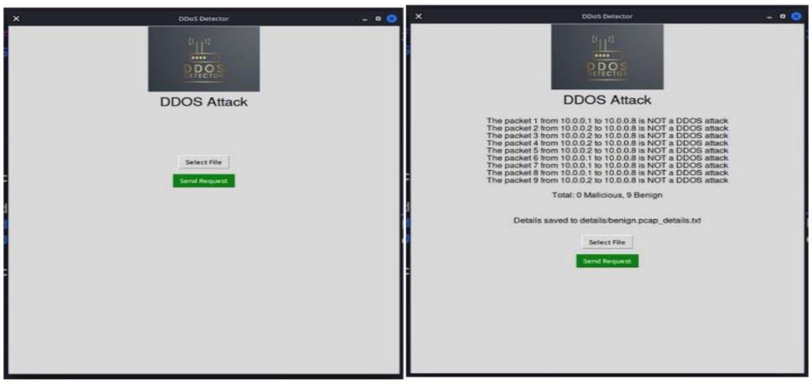
The Tkinter app uses Python's built-in Tkinter library to generate a graphical user interface (GUI) that users can use to communicate with the DDoS detection system. This part doesn't depend on sophisticated ideas or specialized services like neural networks or natural language processing. Rather, it acts as the UI/UX frontend that makes user interactions easier. Users can upload traffic data files, initiate DDoS detection, and see the generated predictions. This UI system serves as a bridge between the user and the program's core functionality, which allows users to start and stop DDoS detection tasks without having to directly deal with the machine learning models or backend processes. It improves accessibility and usability.

RESULTS:

The methodology suggested is composed of three primary parts. Data preprocessing is done first to manage noise, missing values, and class conversion. Machine learning models are then applied using Scikit and Spark libraries for both normal and big data solutions as the second part. The results are compared using evaluation metrics as the last part. Our solution's efficacy relies on the model selection and their hyperparameters. But limitations are there, such as possible information loss in Principal Component Analysis (PCA), since not all the features can be of equal significance. Also, our solution will perhaps not take into account more sophisticated methods and intricate patterns, and it can neglect dynamic behaviors or changes that can influence attack detection. To overcome these limitations, we broadened the preprocessing scope and considered a broader variety of machine learning models. Our project augments the solution by adding further preprocessing steps and implementing logistic regression, SVM, KNN, sequential neural networks, decision trees, and random forests, as well as their hyperparameter tuning.

Our proposed methodology is centered on mitigating the threat posed by Distributed Denial of Service (DDoS) attacks within Software-Defined Networking (SDN). DDoS attacks constitute a significant threat to network security as they have the ability to stall services and induce substantial economic losses. We recommend exploiting the strengths of Network Software-Defined Networking, including in-depth packet analysis and dynamic traffic policy management, to improve DDoS attack detection. Our study addresses some of these traditional network architecture-based DDoS detection techniques such as traffic characteristic-based detection and traffic anomaly-based detection. These techniques are based on building characteristic databases, traffic modeling, and monitoring unusual flow patterns. These techniques may be complicated, delay detection, or have high false positives. Our solution's effectiveness comes from its application of SDN's potential to implement deep packet analysis, traffic management that can be easily adjusted, and rapid response to changes in policy. In addition, our project extends existing research focus on SVM by adding hyperparameter optimization for every machine learning model and evaluating comprehensive metrics. Should our project use real-time detection, it might have faster responses to looming DDoS attacks.

Additionally, our methodology involves designing a machine learning model based on Multiple Linear Regression (MLR) analysis and creating data visualizations through residual plots. The primary purpose is to apply MLR to the CICIDS 2017 dataset, a benchmark dataset commonly used in research. Our process includes feature selection using the Information Gain (IG) approach, where the chosen features are then subjected to MLR analysis. The effectiveness of this solution is contingent on the quality and accuracy with which these linear regression models are trained. But still, this solution is limited because MLR is based on a linear relationship between data points. If the relationship is not linear, then the performance of the model may be drastically impacted. The efficiency of the IG also depends on the fact that the features selected are most relevant to our target variable. Finally, the performance of the solution depends on the quality of the CICIDS 2017 dataset. Contrarily, our project explores methods that deal with nonlinear relationships among features and target variables. Examples include SVM (hyperplane), KNN (distance-based), and decision trees/random forests. These models have a very good capability of identifying complex nonlinear variable interactions, thus improving the overall detection ability of our system.



```
The packet 1 from 10.0.0.1 to 10.0.0.8 is NOT a DDOS attack
The packet 2 from 10.0.0.2 to 10.0.0.8 is NOT a DDOS attack
The packet 3 from 10.0.0.2 to 10.0.0.8 is NOT a DDOS attack
The packet 4 from 10.0.0.2 to 10.0.0.8 is NOT a DDOS attack
The packet 5 from 10.0.0.2 to 10.0.0.8 is NOT a DDOS attack
The packet 6 from 10.0.0.1 to 10.0.0.8 is NOT a DDOS attack
The packet 7 from 10.0.0.1 to 10.0.0.8 is NOT a DDOS attack
The packet 8 from 10.0.0.1 to 10.0.0.8 is NOT a DDOS attack
The packet 9 from 10.0.0.2 to 10.0.0.8 is NOT a DDOS attack

Total: 0 Malicious, 9 Benign

Details saved to details/benign.pcap_details.txt
```

FUTURE ENHANCEMENT:

Looking ahead, there are a few optimizations that can be made to further enhance the DDoS detection system. The first is in the area of real-time monitoring and alerting functionality. Through the implementation of a constant monitor system, the solution can scrutinize incoming traffic patterns in real-time, facilitating instant detection and intervention against possible DDoS attacks. This forward-looking mindset would not only reduce service disruption but also allow network administrators to respond quickly and take measures to counter threats prior to their being a full-fledged threat. Furthermore, using machine learning strategies that evolve alongside changing attack vectors will improve the system's capacity to withstand novel and complex DDoS attacks, keeping it effective in the face of ever-changing threats.

Another direction for future development is widening the scope of machine learning models and methods employed in the detection process. Although our present project has experimented with several models, such as SVM, KNN, and decision trees, it is possible to introduce more sophisticated algorithms like deep learning models, which have proven useful in processing complex data patterns and large data. Methods such as Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) may prove especially useful to examine time-series data and look for complex relations in network traffic. In addition, ensemble strategies that integrate numerous models may help enhance detection effectiveness and minimize false positives, which would be an even more comprehensive solution for detecting DDoS.

Finally, improving the Tkinter application and Flask web program user interface and experience will play a key part in wider deployment of the system. Future upgrade could involve making interactive dashboards to display traffic flows, detection outcomes, as well as trend historical data into visual representations that are easy to interpret. That way, useful insights into an organization's posture in terms of network security could be gained through visualization, enhancing decision-making capability. Further, the inclusion of user feedback processes may assist in streamlining the system's performance and usability, making it fully meet the expectations of network administrators and security professionals. Through attention to these updates, we envision a more enhanced and user-centric DDoS detection solution capable of evolving along with the growing challenges of network security.

CONCLUSION:

In summary, this project presents a robust DDoS attack detection solution. The system achieves an accurate classification process by intricately combining efficient data preprocessing pipelines and a spectrum of machine-learning models. The project also provides two UI/UX the users can choose from, the Tkinter app and the Flask web application. These 2 UI provides a friendly way of interacting with the ML model. One of the limitations of this project was the challenge of capturing real-life data and feeding it into our model. Real-time capture is crucial for identifying potential threats and responding promptly. We should ensure a smooth pipeline from data capture to model input to minimize the overall time. To address this limitation, we could prioritize minimizing the latency in data processing. Techniques such as stream processing could be considered to guarantee a continuous flow of real-time data to the model. While minimizing latency was essential, we recognized the need to strike a balance between speed and accuracy, given there is a tradeoff. Therefore, we could tune hyperparameters based on time to ensure quick classification while still considering accuracy. This includes finding and adjusting parameters that influence both accuracy and prediction speed. Similarly, we could eliminate more features that have relatively less impact on results. In addition to traditional features, we could introduce metrics that track the frequency and nature of interactions between IP addresses and the network. These metrics provided me with a more comprehensive view of network behavior to identify subtle patterns.

REFERENCES:

1. Lau, Felix, et al. "Distributed denial of service attacks." SMC 2000 Conference Proceedings. 2000 IEEE International Conference on Systems, Man, and Cybernetics. 'Cybernetics Evolving to Systems, Humans, Organizations, and Their Complex Interactions' (Cat. No. 0). Vol. 3. IEEE, 2000. *to Biometrics*. Springer Science & Business Media.
2. Rogers, Larry. "What Is a Distributed Denial of Service (DDoS) Attack and What Can I Do About It?" CERT Carnegie Mellon University (2004).
3. Ying, Xue. "An overview of overfitting and its solutions." Journal of Physics: Conference Series. Vol. 1168. IOP Publishing, 2019.
4. Zaroo, Puneet. "A survey of DDoS attacks and some DDoS defense mechanisms." Advanced Information Assurance (CS 626) (2002).

5. Browne, Michael W. "Cross-validation methods." Journal of Mathematical Psychology 44.1 (2000): 108-132.
6. Awan, Mazhar Javed, et al. "Real-time DDoS attack detection system using big data approach." Sustainability 13.19 (2021): 10743.
7. Lima Filho, Francisco Sales de, et al. "Smart detection: an online approach for DoS/DDoS attack detection using machine learning." Security and Communication Networks 2019 (2019): 1-15.
8. Ye, Jin, et al. "A DDoS attack detection method based on SVM in software-defined networks." Security and Communication Networks 2018 (2018).
9. Hoque, Nazrul, Hirak Kashyap, and Dhruba Kumar Bhattacharyya. "Real-time DDoS attack detection using FPGA." Computer Communications Networks 110(2017): 48-58.
10. Xu, Yang, and Yong Liu. "DDoS attack detection under SDN context." IEEE INFOCOM 2016 – The 35th Annual IEEE International Conference on Computer Communications. IEEE, 2016.