Detection of DDoS Attack on Smart Home Infrastructure Using Artificial Intelligence Models

Thejavathy Vengappa Raja
Dept. of Computer Science
Nottingham Trent University
Nottingham, United Kingdom

Zoheir Ezziane
Dept. of Computer Science
Nottingham Trent University
Nottingham, United Kingdom
zoheir.ezziane@ntu.ac.uk

Jun He
Dept. of Computer Science
Nottingham Trent University
Nottingham, United Kingdom

Xiaoqi Ma
Dept. of Computer Science
Nottingham Trent University
Nottingham, United Kingdom

Asmau Wali-Zubair Kazaure
Dept. of Computer Science
Nottingham Trent University
Nottingham, United Kingdom

Abstract—The whole web world is concerned and constantly threatened by security intrusion. From the topmost corporate companies to the recently established start-ups, every company focuses on their network, system, and information security as it is the core of any company. Even a simple small security breach can cause a considerable loss to the company and compromises the CIA Triad (Confidentiality, Integrity, and Availability). Security concerns and hacking activities such as Distributed Denial of Service (DDoS) attacks are also experienced within home networks which could be saturated reaching a crashing point. This work focuses on using Artificial Intelligence (AI) and identifying suitable models to train, identify, and detect DDoS attacks. In addition, it aims to implement on smart home datasets and find the best model from those which performs with a high accuracy rate on the smart home dataset. The novelty of this project is identifying one best AI model among many of the existing models that works best on smart home datasets and in identifying and detecting DDoS attacks.

Keywords—smart homes, DDoS, machine learning, deep learning, AI, cybersecurity

I. INTRODUCTION

Everything is moved to digital and smart technology. Digitization has brought massive economic growth on one side and as with every emerging technology there comes risks on the other side. Confidentiality, integrity, and availability are part of what is known as the CIA triad. It is a model designed to guide policies for information security within an organization [1]. If any one of these core principles is compromised, then there is a considerable risk to be faced and a great chance to become a target of attacks. Cybersecurity is facing a colossal threat in this modern technology. There are so many types of security threats on the web, but the Distributed Denial of Service (DDoS) attack is described as the most disastrous attack which is very much capable of making the system unavailable to its actual users to destroying the resources completely [2], and during this pandemic period as remote working became the trend, DDoS attack is one of the 5 topmost threats to the networks [3].

A smart home is also labelled as “connected homes”, “Home Automation”, or “Intelligent home” [4]. The smart devices or systems installed or used in the home function using artificial intelligence (AI), adapts and assists according to the individual home/personal needs or lifestyles [5]. Smart homes are basic devices or appliances set up in a network as an architecture in a home or building that are used to control, assist or detect actions, and functionalities using AI automation. These smart technologies are not only for homes but also for a wide range of different sectors.

In this research, we are focusing only on smart home architecture. It is basically the smart devices including washing machines, lights, cameras, and other home appliances connected under one network to perform certain functionalities or tasks by bringing ease in day-to-day life and improving efficiency.

Any hacker has three basic ways to violate the target system [6]:

1. Find a way to seek into the private information space - compromising ‘Confidentiality’.
2. Get access to alter the confidential private information - compromising the ‘Integrity’.
3. Make the service unavailable for its actual users -compromising the ‘Availability’.

The third method is known to be the most common attack which can be created by a novice hacker without any admin privileges. On the other hand, the first two methods are more sophisticated to carry out. The DoS/DDoS attack uses the third way which uses one or more bots to compromise the availability of the target system. The DDoS attacks are widely classified into three major types [7]:

- Traffic/Fragmentation attack
- Bandwidth/Volume attack
- Application attack

As DDoS is one of the major threats to cybersecurity, researchers are focusing on implementing the typically trained AI systems to prevent evolving DDoS attacks. The first-phased trained systems have already been implemented to detect and mitigate the attacks. Attack detection using AI is nothing new to the technology including related techniques [7].

Even with the most advanced prevention techniques and standards, the attackers find a way to attack. The need for DDoS attack detection and mitigation is more significant than anything as the attacks are becoming fiercer. Hence, there is a need for dynamic enhancement in techniques of prevention, detection, and mitigation which can be accomplished using AI [8]. Techniques like semi-supervised learning and self-supervised learning and other tools help to achieve this. There are many AI models and techniques available that are used to detect and mitigate DDoS attacks. A proper study of available techniques and identifying the best model to detect and mitigate DDoS attacks, and to adapt to modern attacks, is required even though the attacks are being newly evolved. Therefore, using AI would be a weapon to prevent, detect and mitigate DDoS attacks. There are many benefits of applying AI in cybersecurity and particularly in DDoS attacks [9, 10] such as better and faster detection, maintaining high-level authentication, performing repetitive security processes or monitoring tasks, and monitoring and performing behavioural analytics.

These functionality benefits of AI are very useful for securing systems from being attacked by the most devastating attacks such as DDoS attacks. Thus, this work focuses on implementing AI to secure smart homes.

There is a prominent need in identifying the best AI method to prevent, detect and mitigate DDoS attacks on smart home networks even during advanced attacks. Modern technology is facing modern attacks. We need a solution which can adapt and sense any sort of attack at an early stage. Much research is being held on implementing different AI methods to detect a DDoS attack at an early stage in general. This work aims to identify one best AI method among the top methods which can detect DDoS attacks on smart home networks more efficiently and be able to dynamically adapt in identifying zero-day attacks.

II. Related Work

DoS attacks have evolved into DDoS attacks over time. Currently, these DDoS attacks are becoming one of the crucial, challenging, and fast-evolving threats to the Internet world [11]-This attack is basically causing the target system to exhaust by responding to an active loop of dummy requests making it unavailable to its actual users. These hacked systems under control are called bots or zombies. A network of zombies or bots is called a botnet. DDoS uses a large network of botnets to create an attack and suspend the target system from performing or serving to its actual requests [12]. Basically, a DDoS attack involves or requires two major conditions to perform: (1) A malicious packet; and (2)-A botnet.

The following are a few common types or techniques of DDoS attacks described in brief [13] including Smurf Attack, UDP flood; HTTP flood Attack; Teardrop attack; Point of death attack; SYN flood attack; and Buffer overflow attack. There are a few general techniques in mitigating the DDoS attacks which are being used by various organisations [13]. These include blackholing, routers, firewalls, IDS, signature detection, anomaly detection, rate limiting, and anycast network diffusion.

According to [14], modern DDoS attacks are widely using two basic trend concepts.

- Largest volumetric attack and highest intensity flood: The speed, volume and size of the attacks are all becoming bigger recently.
- Multi-vector DDoS attacks: Different types of attacks combined as an attack are the most crucial one and it is very hard to be detected. The mitigation is diverted to one attack type while another type just does its work.

The attackers are moving towards utilizing the emerging popular technology – AI to create a malicious attack. Where it helps the attackers to mostly automate the attack and creates the most defective crucial attack with severe consequences and shuts the system down within less time. The AI DDoS attacks are gradually increasing as it provides the hacker more benefits rather than creating it manually [12]. Even with the growing technologies like 5G, IoT, and smart systems, the threat is immense.

According to Khalaf et al. [12], there are different AI methods that can be used to detect attacks. Chidananda, Murthy, and Madhu [13] describe the DDoS attack in Cloud computing and give a basic outline of preventing it using machine learning (ML). They focus on highlighting the Artificial Neural Network (ANN) method by demonstrating the architecture describing the proposed system to prevent DDoS attacks by monitoring system resources and traffic at various levels and filtering the traffic. They also propose a theoretical system for a neural network to prevent DDoS attacks, without testing or examining any proposed work on this approach. Their work highlights that this system is only focused on monitoring the traffic.

Zhang, Zhang and Yu [15] and Glăvan et al.[14] reported the most prominently and frequently used techniques including Bayes classification, ANN, and support vector machines. Other than the AI methods implementation, Alzahrani and Hong [16] focus on anomaly and signatures of the traffic and propose a joint anomaly and signature-based detection using AI. Both the accuracy and detection rate of the joint anomaly and signature-based detector using an integrated ANN are higher and almost close to exact detection with 0.00% of false positives.

Said, Overill and Raszik [17] proposed an ANN algorithm to separate the actual traffic and malicious traffic where it identifies known attacks with 100% accuracy and unknown with 95%. Their approach gives overall 98% accuracy which is greater than other techniques like snort and PNN. The reported ANN model with its outstanding scope, dynamic adaptability, and exponential characteristics, can be implemented to prevent and detect DDoS attacks.

The smart home is a heterogenous network with different devices and computational capabilities and its dynamic nature can easily be vulnerable to various threats. Lymberopoulos and Komninos [18] categorize security into two main categories: Internal threats and External threats.

The main security concerns include unauthorized access, data privacy, applications or devices limitations, different level of security requirements for different devices., arbitrary wireless devices, an internal network connection to an external network, and a static IP address for the internal network.

Recent research [19] describes the security issues with smart home applications and provides security techniques as a solution. In those listed technologies, ML algorithms and ANN algorithms are suggested. Dalal, Tushir and Dezfouli [20] provide a quantitative study that describes the DDoS and energy-oriented DDoS (E-DDoS) attack on smart homes and analyses each component of victim like payload, port states, protocols, and attack rates. Saxena, Sodhi and Singh [21] describe the fact of a smart home that has an open embedded operating system which sometimes represents a way to get through for hackers and provides a solution to DDoS attacks on smart homes.

Gordon et al. [22] propose an environment and a stateless flow-based feature for detecting DDoS and device classification. This proposed feature helps ML models to detect and classify with better accuracy. They experiment on 3 ML models: KNN, LK-SVM, RF. The study of each existing literature has given the idea that there is no specific existing model that is suitable for smart homes. Only a few may have mentioned ANN as a better approach theoretically. So as a novel approach we will be implementing ANN along with KNN and then find which one is more efficient. As a novel approach, we are going to compare the ML classification model with the deep learning model.

III. Methodology

In this work, we are using the CRoss Industry Standard Process for Data Mining (CRISP-DM) Methodology for data mining and data analysis. He [23] explores CRISP-DM method along with a case study implemented on a dataset and analyses the data for ML. It is a standard model process that is a data-science life cycle with six phases. This model helps ML projects in properly planning, organizing, and implementing data mining and data analysis. It is kind of a software development life-cycle process model and includes the following six phases: (1) Business Understanding; (2) Data Understanding; (3) Data Preparation; (4) Modelling; (5) Evaluation; and (6) Deployment.

A. Data Understanding/Collection

Collect Initial Data: The initial data is collected from smart home network traffic as benign data and attack data [24]. Using Wireshark, the network traffic data is collected and saved as a .pcap file and .csv file. The normal smart home network traffic data is collected in 'NormalData.csv' file and the DDoS-generated smart home network traffic data is collected in 'DDoSData.csv' file. The csv file is required to implement the project so the .pcap file /traffic is saved to .csv file without any data loss.

Describe data: Figure 1 illustrates the smart home as it is set up with a bulb, plug and motion sensor. In this setup, to experiment with the DDoS attack, the LOIC (Low orbit Ion Cannon software) tool is used. To conduct this experiment, an open-source DDoS attack tool was used by ethical hackers or penetration testers to check and test the security and stress level of the network. LOIC is used to conduct attacks on the smart home static IP addresses. Even then Wireshark highlighted DDoS attack (SYN Flood) traffic in its network capture. The data collected during an attack is used in the project as attack data and the general traffic is the normal smart home dataset which is benign data. The DDoS attack architecture is shown in figure 2, in which three laptops with LOIC targeting the static IP address of the smart home hub. Thus, the attack data is collected using the Wireshark network traffic analyser.

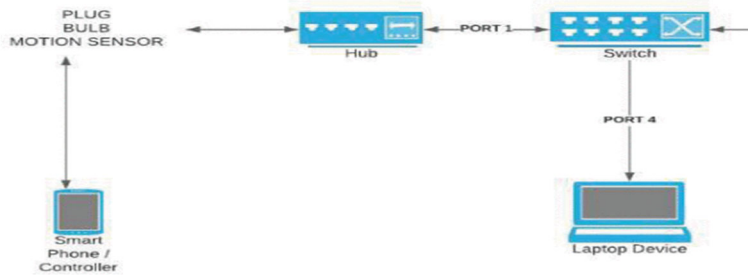

Fig. 1. Smart Home Architecture [25]

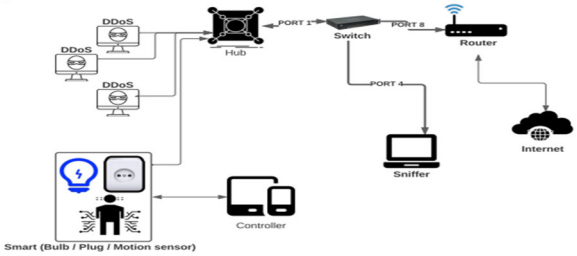

Fig2. DoS attack on smart home architecture [24]

To explore data in depth, the Exploratory Data Analysis (EDA) is performed before processing ML algorithms on the data. The dataset quality is assessed when converting it to .csv from Wireshark, which is as same as the network traffic. The feature values format is different from the other common dataset values like NSL-KDD, because it is just the raw data and needs to be processed and converted to the required format. It can also be conducted in the data pre-processing phase.

B. Data Preparation

This phase can be technically called the 'Data Pre-processing' phase. This is the most important phase in any ML project. Scikit-learn data pre-processing is used generally to convert the raw dataset features containing missing or unusable values into standard required feature values that are suitable for ML models to learn. The steps involved in data pre-processing include: (1) Import the required libraries; (2) Import the data set; (3) Handle the missing data; (4) Encode categorical data; (5) Splitting the data set into test and training set; and (6) Feature scaling. The last step in data pre-processing is about normalising data. After converting all the data to standard numerical values, the data needs to be prepared for the model, as all the values are in numbers where the range may vary enormously, so to bring all the values into one common scale without affecting the original range.

In splitting the data, the whole data is split into training and testing sets. The data in the training set is used to train the ML algorithm model. To test the proper accuracy of the model, the test set can be used to predict the outcome of the experiment. But in this work, the training set represents

80% and the testing set 20%. As we are dealing with DDoS attack data, the model needs to be trained well to implement functionality properly.

C. Modelling Phase

Select modelling Techniques: There are many AI models that are available with each individual speciality and functionalities. Based on the literature review, the suitable models for this project are ANN and KNN. The type of AI considered here is ML versus deep learning and supervised versus unsupervised learning. So, the KNN is selected for implementing a model in ML and ANN for implementing a model in deep learning. As discussed in the literature review, KNN is chosen on the other side to learn how it classifies the data, as the KNN is based on classification and the data just has two labels to classify and it can categorize using nearest features.

1) K-Nearest Neighbour

This algorithm looks simple yet it is one of the major classification ML models. It sets a boundary based on characteristic values and classifies data accordingly with the nearest falling group. The value K decides the grouping [26] It is also known as a non-parametric algorithm which uses proximity to classify or predict the grouping of a data point. Thus, this model is suitable for identifying DDoS attacks, even with a slight detection of similarity the model categorizes it to attack data including the least possible one. The model consists of two important processes: Selecting the K value from the available neighbours and calculating Euclidean distance, which decides the quality of the model. KNN has two major steps: Learning Step (training) and Classifier Assessment Step.

ANN works like the human brain's neural system, and it consists of nodes [27]. ANN is mainly of two types: Feed forward neural network and feedback neural network. In this work we are using the feedforward neural network which means the neural network processes the information and forwards it to the next node and the data is processed in a unidirectional flow [28].

D. Evaluation Phase

The models are critically evaluated. The performance of both models is evaluated using a few standard metrics as follows:

- Confusion Matrix: The confusion matrix identifies the false positives (FP) and False Negatives (FN) along with True Positives (TP) and True Negatives (TN) which play an important role in defining the efficiency of a model.
- Precision: It is defined as the ratio of actual positives from total predicted positive values.
  $$Precision = \frac{TP}{TP+FP} = \frac{Predictions\ Actually\ Positive}{Total\ Predicted\ Postive}$$
- Recall: It defines the actual correct positive predictions.
  $$Recall = \frac{TP}{TP+FN} = \frac{Prediction\ Actually\ Postive}{Total\ Actual\ Postive}$$
- F1 -Score: It is an average of precision and recall.
  $$F1 - Score = 2 * \frac{(Recall\ *\ Precision)}{(Recall + Precision)}$$
- Accuracy: It is the overall true predictions both positives and negatives.
  $$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} = \frac{Correct\ Predictions}{Total\ Predictions}$$
- FPR and FNR: This is the false positive rate (FPR - the type I error) and the false negative rate (FNR- the type II error) is calculated as follows:
  FPR = False positive / Total Negative
  FNR = False Negative / Total Positive
- ROC Curve: It means the Receiver Operating Characteristic curve. It is an important graph that is used to visualize the performance of the classification model at all thresholds. This graph consists of two factors: True Positive Rate (TPR) and False Positive Rate (FPR). It is said that the model efficiency is defined by the curve area.

E. Deployment Phase, Tools and Libraries Used

The implementations are documented and reported. The plan is monitored to avoid any major issues.

The device used for this research project is a DELL laptop which is a windows 10 operating system with 64-bit OS, x64- based processor, 8GB RAM and Intel(R) Core (TM) i5- 7200U CPU @ 2.5GHz 2.70GHz processor.

The programming language used for this project is Python and it is widely used in ML research projects. Google Collaboratory (Google colab) is used as a development tool/platform. Wireshark is used to monitor and analyse the traffic and to extract the csv file. Python libraries used are as follows: NumPy – to perform mathematical calculations; Scikit-learn – to perform all the ML processes and analysis; Pandas – to handle datasets, and Matplotlib – to plot and visualize data.

IV. Implementation & Results

AI models are implemented on the collected smart home dataset to see the performance of each model in identifying the DDoS attack.

A. Normal Dataset, DDoS Dataset, and EDA Analysis

The normal traffic as captured in the Wireshark is collected and converted to .csv file. This .csv file consists of the normal dataset which is benign data. The normal dataset includes the following features, No.: The traffic sequence number; Time: Time stamps for each traffic flow or network conversation; Source: Source Id of each current traffic flow; Destination: Destination Id of each current traffic flow; Protocol: Describes which protocol is being used for communication; Length: Length of the data packet; Info: This describes the state of the network conversation in detail.

The normal dataset contains 22,296 data entries in 7 columns which are features. The normal dataset contains traffic flow for approximately 10 hours and 45 minutes.

Similar to the normal dataset, this dataset also has the same feature columns. The Attack data set contains 1,00,014 entries in 7 columned features. This data is captured for approximately only 42 minutes.

The EDA helps to give a better understanding of the data that even the smart home dataset – both normal and DDoS traffic data - can be analysed and understood using EDA. This helps to analyse the traffic flow only for 1 minute. In this work, some functions of EDA are implemented to understand both datasets before data pre-processing.

B. Data Pre-Processing

Before data pre-processing, labelling data is required as there are two classification data – Normal and DDoS data. Normal data is labelled 0 and attack data is labelled 1. DDoS and Normal dataset have added a column called label with their specific labels. The entries in each column are in its original format, which is raw, as the model requires it all to be in specific ranged numeric values, which needs to be converted to.

1) Raw Smart Home Data Conversion

Label_Encoder: All the columns need to be encoded to standard values without losing data except the number column. The time is converted to a Unix timestamp. As the source and destinations are in IP address format, they are converted to numerical characteristics. Each IP address is assigned a number. The final one that needs to be label encoded is protocol..

The labels assigned to each protocol used in our ML model are as follows:

- When assigning labels to the protocol, the following output is given. To identify the protocol assignment. The counter function is used to calculate the count of total protocols before and after labelling. Therefore, DHCP, DNS, ICMP. NTP, and TCP are assigned labels 0, 1, 2, 3, and 4 respectively.
- Protocols are labelled with numbers while processing data uses a label encoder. The scikit-learn has a pre-processing library in which the Label encoder function is available.
- le = LabelEncoder(); Encoded_values = le.fit_transform(column1 values)
- Importing the label encoder function and passing the required column values to the label encoder object along with fit_transform function converts all the values and assigns them to the specified variable.
- Filtering features: The info section of the dataset is not required at this stage. So, we are clearing the unnecessary data at this point by dropping the column.

Only the required features are being carried forward to the next process.

Feature scaling: Data is converted, in which the length and most importantly time values are in a major range difference from the rest. To bring data uniformity, as the models require data to be in similar ranges. The whole data is converted to a specific range without affecting the values. We are using normalization at this step to do scaling which brings the data range to 0 and 1 by using min/max. The scikit-learn pre-processing library includes the normalize function which scales numbers into the range.

Normalized_df = preprocessing.normalize(df)

Splitting Dataset: After data pre-processing, the main process is to split the dataset into train and test set so that the model can train and learn from the allocated training dataset and the testing set is used to predict, test or sometimes to do cross-validation testing to evaluate the efficiency of the model.

C. Implementation of AI models

After splitting the dataset to train and test the data sets. The ML model needs to be trained with the training data. As we chose KNN and ANN as our AI models to work on in this research. The below sections show each model implementation.

1) K-Nearest Neighbours (KNN):

In this model the KNN is executed on the integrated smart home dataset containing benign and attack data, it learns and trains the model with the available data. The n_neighbors (K) is chosen to be 5 and the metric value is chosen to be 'Euclidean' as it is the most popular one and works well with this dataset.

2) ANN:

This model is chosen because it is well known for recognizing patterns and for classification in 'Deep Learning'. The smart home dataset is trained through this model to learn the patterns and characteristics along each layer.

We have created the ANN model network with 3 layers to start with. The number (No) of neurons in first two layers is 6. The input_dim is 6 as we have 6 features in the dataset: No, Time, Source, Destination, Protocol, Length (50,51). The kernel_initializer is uniform as it generates the uniform weights on each link. The Rectified linear activation function (ReLU) is used in first two layers to process fast and efficiently. In addition, the output layer has sigmoid function as it is a non-linear function it gives value only between 0 and 1. 'Adam' optimizer is used as best properties to optimize the model and for loss parameter 'binary_crossentropy' is used as our labels are binary values. The model is set for 10 epochs at first to avoid overfitting the model.

V. Results and Discussions

A. Implementation Findings

1) KNN:

The error rate, the accuracy rate, and the ROC curve for the KNN model are depicted in figures 3, 4, and 5 respectively.

Figure 3 indicates the error rate for the KNN model, graph has two axis error rate and K value. The error rate is under 12% which is good at first glance. It's the training error rate for the KNN model. Figure 4 shows the accuracy rate for the KNN model, the accuracy rate measured with the k value. The accuracy rate is 98% because it is just training set accuracy and as the dataset is small. Figure 5 provides the ROC curve in a graphical representation to see the performance of the KNN model.

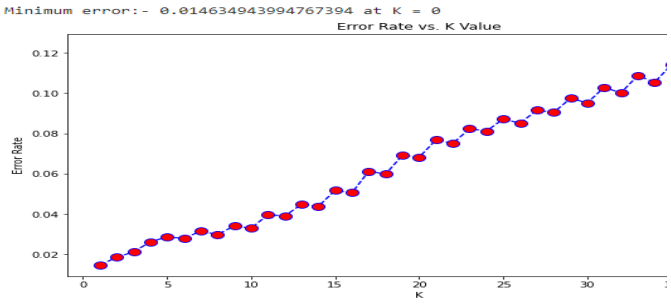

Fig. 3. Error rate for the KNN model

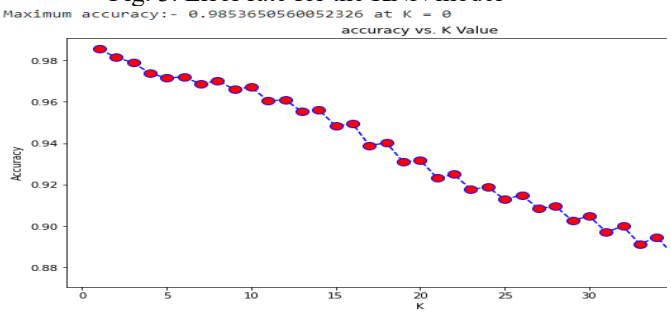

Fig.4. Accuracy rate for the KNN model

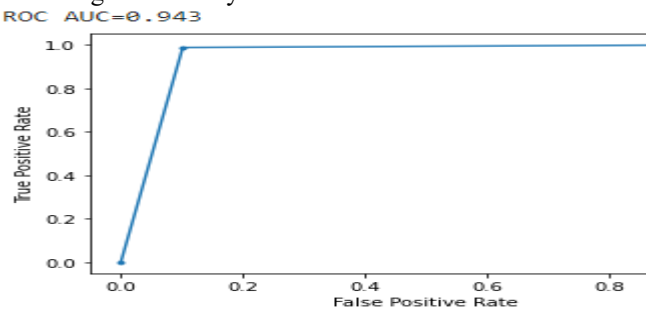

Fig. 5. ROC curve for the KNN model

2) ANN:

The ANN was tuned and tested to improve the efficiency. 2s 3ms/step - loss: 0.4729 - accuracy: 0.8191. In the second trial the no. of neurons in each layer are modified as follows: 12, 8, 1. Accuracy and loss for the ANN model are illustrated in figures 6 and 7 respectively.

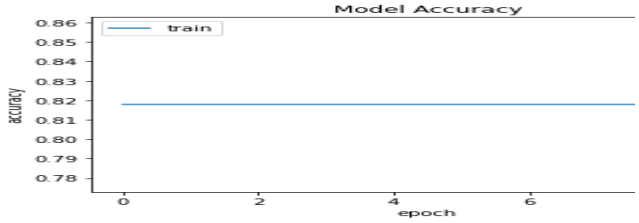

Fig. 6. ANN model accuracy

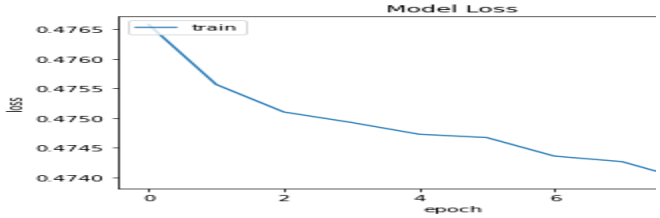

Fig. 7. ANN model loss

B. Critical Evaluation and Discussion

1) Performance Evaluation

The main performance metrics to consider in this work is model accuracy and loss scores as shown below:

Model	Accuracy score	Loss score
KNN	0.99	0.015
ANN	0.82	0.47

The ANN has 82% accuracy and KNN has 99%.

2) Overall Evaluation

In KNN the confusion matrix has more True Positives and True Negatives in prediction. Whereas in the ANN the True Positives are higher, True Negatives and False negatives are 0 which describes that it needs more data to learn to classify between DDoS and non-DDoS data, everything is classified as a DDoS which is not ideal. Even after changing the parameters and tuning the model, it always resulted in only 82% accuracy which is not bad to begin with a small dataset yet looking at the false positives and without negative classification, however, it needs to be more efficient. The data is split into 80 and 20 even with this small dataset; however, it may not be enough for ANN.

It is interesting to note that in this work, KNN works better than ANN for this particular dataset. As for now we just need the pattern of attack to make the model learn, so we focused on collecting data at different circumstances – The usual traffic flow and under a DDoS attack.

The KNN model experienced no issues in training the dataset. With less loss score it showed a good accuracy score which makes the model look better and more efficient at this stage. On the other hand, training the ANN model was a bit of a hassle and the accuracy did not increase during the experiment. To avoid overfitting, 10 epochs were tried in the beginning as it showed 82%, even after retuning with the parameters it showed the same results. Consequently, the epochs were increased to 50, where each epoch took so long, and it became slow, even then each epoch showed the same 82% for 50 epochs, there was no improvement. Even after all these trials the ANN was just 82% and the loss, TP, TF, FP, FN are also ineffective which make the model look weak. Even the f1-score was very low for ANN, whereas the KNN's f1-score, precision and recall were good for both classes as shown below:

	precision	recall	F1-score	Support
0	0.94	0.90	0.92	4458
1	0.98	0.99	0.98	20004

VI. Conclusion

According to the above findings, the KNN is the best model for the available smart home dataset at this point with 99%. The implemented ANN is found to be very less efficient at this point due to its ineffective results. In this case, we might also consider that the smart home data collected is limited and not as important as the other benchmark dataset and as this dataset is only the collection of traffic flow data for 10 hours and 45 minutes. In addition, we only considered 6 features for this dataset avoiding overfitting. This consideration will not be sufficient to train the models, decide and reach final conclusions proving that a model is the best AI model. Eventually, those remarks and suggestions would help to propose a mitigation technique for DDoS attacks after training a model for greater accuracy and efficiency.

It is generally reported that KNN is the best for a certain level, small-scaled normal datasets, whereas the ANN is suitable for big datasets, and that could be the reason why it worked here as well and reached similar results.

For future work, this smart dataset will be used for collecting vast traffic flow, and probably using ANN would be more efficient. Hence, considering the aim and objectives of this research with the available limited dataset, and at this stage of research, the KNN is the best AI model to classify and identify the DDoS attack on the smart home network.

REFERENCES

[1] MIR, S. and QUADRI, S. Information Availability: An Insight into the Most Important Attribute of Information Security. Journal of Information Security, 07, 2016, pp. 185-194.

[2] Devi,B. S, Preetha, G, Selvaram, G. et al. An impact analysis: Real time DDoS attack detection and mitigation using machine learning. In: - 2014 International Conference on Recent Trends in Information Technology,2014, pp. 1-7.

[3] Gurinaviciute, J. 5 biggest cybersecurity threats, Security Magazine. 2021.

[4] Lupton, D., Pink, S. and Horst, H. Living in, with and beyond the 'smart home': Introduction to the special issue. Conv., 27 (5), 2021, pp.1147-1154.

[5] Alaa, M., et al. A review of smart home applications based on Internet of Things. Journal of Network and Computer Applications, 97, 2017, pp.48-65.

[6] Tripathi, N. and Mehtre, B. DoS and DDoS Attacks: Impact, Analysis and Countermeasures. 2013, pp. 1-6.$

[7] Sambangi, S. and Gondi, L. A Machine Learning Approach for DDoS (Distributed Denial of Service) Attack Detection Using Multiple Linear Regression. MDPI AG. 2020.

[8] Basu, V. An approach to detect DDoS attack with A.I. Towards Data Science. 2020.

[9] Martin, D., 8 Benefits of Using AI for Cybersecurity. Cyber management. 2021.

[10] Anon, a. Impact of Artificial Intelligence in Cyber Security. Data Flair.

[11] GMBH, L. and Grave, K. Why Automation and AI are Critical in DDoS Mitigation - Link11. Cyber Security, 2020.

[12] Khalaf, B.A., Mostafa, S.A., Mustapha, A. et al., Comprehensive Review of Artificial Intelligence and Statistical Approaches in Distributed Denial of Service Attack and Défense Methods. 2019.

[13] Chidananda AJ., Murthy P, Madhu BR. Prevent DDOS Attack in Cloud Using Machine Learning. Int J. Adv. Res. in Comp. Sci. and Soft. Eng. 2016; 6(6): pp575-579.

[14] Glavan, D., Racuciu, C., Moineco, R., et al., DDoS detection and prevention based on artificial intelligence techniques. Scientific Bulletin "Mircea Cel Batran" Naval Academy, 2019, 22 (1), pp. 1-11.

[15] Zhang, B., Zhang, T., and Yu, Z., DDoS detection and prevention based on artificial intelligence techniques. In: - 3rd IEEE International Conference on Computer and Communications (ICCC), 2017, pp. 1276-1280.

[16] Alzahrani, S. and Hong, L. Detection of Distributed Denial of Service (DDoS) Attacks Using Artificial Intelligence on Cloud. In: - IEEE World Congress on Services (SERVICES), 2018, pp. 35-36.

[17] Saied, A., Overill, R.E., and Radzik, T. Detection of known and unknown DDoS attacks using Artificial Neural Networks. Neurocomputing, 2016, 172, pp. 385-393.

[18] Mantas, G., Lymberopoulos, D., and Komninos, N. Security in Smart Home Environment, IGI Global. 2011.

[19] COBOI, A., et al., Problems in Smart Homes., 2021, 9.

[20] Dalal, Y., Tushir, B. and Dezfouli, B. A Quantitative Study of DDoS and E-DDoS Attacks on WiFi Smart Home Devices. 2020.

[21] Saxena, U., Sodhi, J.S. and Singh, Y. An Analysis of DDoS Attacks in a Smart Home Networks, 2020, In: pp. 272-276.

[22] Gordon, H., et al/ Securing Smart Homes via Software-Defined Networking and Low-Cost Traffic Classification. 2021.

[23] HE, J. CRISP-DM and Case Study 1. COMP20121 Machine Learning for Data Analytics. 2021.

[24] Wali, A., et al., A novel approach to identifying DDoS traffic in the smart home network via Exploratory Data Analysis. 2022.

[25] Wali, A., et al., An Exploratory Data Analysis of the Network Behaviour of Hive Home Devices. 2021, In: pp. 1-8.

[26] Guo, G., et al., KNN Model-Based Approach in Classification. 2004.

[27] Jeswal, S.K. and Chakraverty, S., 2021, Chapter 10 - Fuzzy eigenvalue problems of structural dynamics using ANN. In: S. Chakravarty, ed., New Paradigms in Computational Modelling, and Its Applications. Academic Press, 2021, pp. 145-161.

[28] Mhatre et al. A Review paper on Artificial Neural Network: A Prediction Technique. 2015.