

Declaration		
Name 1	Percentage 1	Work items completed by 1
Name 2	Percentage 2	Work items completed by 2
Name 3	Percentage 3	Work items completed by 3

Fig. 1. Contribution Declaration

Parkinsons model prediction using data mining methodologies

Abstract—Parkinson’s Disease is a neurodegenerative disorder of dopamine systems. It can be treated but the effectiveness of pharmacological options faces diminishing returns and anything that might reduce the occurrence of overmedication would have strong positive impact. [PREVIOUS PAPER] collected data comparing features of speech in 52 individuals diagnosed with Parkinson’s and analyzed it using deterministic linear regression. In this work, we used various Data Mining tools to prepare and analyze the data speculatively in order to discover interesting information for use in Machine Learning or other subsequent analysis. K-means methods showed [WHAT k-MEANS SHOWED] while OPTICS showed [WHAT OPTICS SHOWED]. DBSCAN was not viable due to [REASON NOT VIABLE]. Validation of clustering was performed using [METHOD FOR VALIDATION]. Classification after Data Mining was performed using SVM [SPELL THAT OUT]. We also approached the problem as an Association Rule problem and discovered [WHAT DOES IT MEAN]?

Index Terms—Data Mining, k-means, k++, OPTICS, DBSCAN, Parkinson’s Disease, SVM

I. INTRODUCTION

Parkinson’s Disease is a neuro degenerative disorder. It is usually diagnosed around 60 years of age and at onset typically presents with tremors and cognitive difficulties. It is not presently known what causes the initial damage, but once the Parkinson’s Disease diagnosis is made, it continues until the patient’s death.

An immense amount of research has gone into understanding and treating the disorder. Postmortem studies have discovered neurological damage begins in the substantia nigra and spreads to the basal ganglia. These two areas are known to use the dopamine neurotransmitter to regulate intentional muscle movement. As the damage progresses, the patient eventually loses all ability to control muscular function.

There are pharmacological treatments that can mitigate the symptoms of the disorder. They work by increasing the amount of dopamine available in the system, or increasing the neuronal sensitivity to dopamine. This is far from an exact science and every drug comes with a limited efficacy period as the systems adapt and damage progresses. Any increase in the efficacy period would bring additional relief to hundreds of thousands. In addition, overmedication can and does occur because the time between administering the drug and the resulting effect can take weeks. This brings new dangers as increasing the available dopamine too far results in the patient expressing schizophrenic symptoms.

Because of this, patients are monitored closely and clinicians are trained to take physiological measurements that are used to determine pharmacological treatment schedules. That

requires direct contact with the patient and is a significant part of the cost of treating the disorder. A Tsanas, MA Little, PE McSharry, LO Ramig (2009) investigated using deterministic regression analysis of 16 components of speech to attempt to predict the clinician’s assessment. These components include things like forms of jitter and shimmer.

In this paper we use non-deterministic methods to investigate the same problem space. We used two general approaches. On one side we used clustering techniques to discover interesting points to use as training models for machine learning algorithms. On the other, we re-imagined the data as transactions and did pattern analysis.

II. RELATED WORKS

Attack detections in IoT systems are notably different from the existing mechanisms because of the special service requirements, such as low latency, resource specificity, distributed nature, mobility, to mention a few. The paper presented by author Satish Pokhrel et.all [2] Proposes an algorithm to detect and mitigate botnet-based distributed denial of service (DDoS) attack in IOT network, in which the authors implement re-sampling Technique called SMOTE (Synthetic Minority over-sampling Technique) over the imbalanced BoT - IoT Dataset. The paper presented by author Jiyeon Kim et.all [4] focuses on developing botnet based detection model for various IOT devices such as Doorbell, Baby Monitor, Webcam etc using various ML(Naive- Bayes, KNN, Logistic Regression, Decision tree, Random Forest) and DL (CNN, RNN, LSTM) models, where the model’s performance is evaluated based on the F1-score by carrying out multiclass classification, as well as binary classification, for each model. The authors use the N-BaIoT Dataset.

The paper presented by Rishabh Gandhi [6] focuses on comparing the various ML and DL models such as, Logistic Regression, KNN, Naive Bayes, Decision Tree, Random Forest, Multi-Layer Perceptron Neural Network, and Long-short Term Memory(LSTM) on Iot Botnet datasets. The author uses three different Iot Based Datasets namely N-BaIoT, IoT-23, Kitsune Network attack Dataset, and MedbIoT Dataset. It is observed that Decision tree performs the best for all the three datasets with less testing and training time.

The paper presented by Yan Naung Soe et.all [8] focuses on developing a ML -Based botnet attack detection framework with sequential detection architecture. The ML models used are ANN, J48 Decision tree, and Naive Bayes and the objective is to detect botnet attacks. As a result of the experiment

conducted the system may need only 1.75 milliseconds to detect a malicious pattern in each of the sub-engine. The results show that ANN and J48 Decision tree algorithms are significantly better than Naive Bayes Algorithm.

In 2022, Mohammed M. Alani proposed an IOT level system that aims to detect botnet presence by examining the packets that flow through it. Traditional intrusion detections were based on network flow, which contributed to performance overheads. The system extracts vital features from incoming and outgoing network packets which forms the basis for detection. The features are mostly focussed on packet-level and detections are primarily based out of classifier ML algorithms [1]

The work depicted in [3] focussed on not only detecting the malicious traffic but also to find the category of IoT Botnet attack. To achieve this, various Machine Learning testers were used out of which, the prominent classifiers include J48, Random Forest (RF) and Multilayer Perceptron (MLP) networks. The approach towards achieving this goal encompasses the process of Synthetic Minority Over-sampling Technique (SMOTE). Best accuracy rates were achieved for the main attack feature and a little less accuracy was obtained in predicting the sub-category of the attack.

Christopher D. McDermott et al. [4] proposed a system utilising Deep Learning algorithm named Bidirectional Long Short Term Memory based Recurrent Neural Network (BLSTM-RNN) and is compared to LSTM-RNN for the purpose of detecting four primary attack vectors. They demonstrated a progressive achievement although the model actually adds performance overhead to epochs such as time. They have also aimed to generate a new comprehensive dataset that includes all ten attack vectors, correspondingly working towards detecting the botnet attack using existing signature and flow based anomaly attacks. A similar Deep learning approach was taken towards identifying the DDoS attacks while ensuring usage of Lightweight DL algorithms by Eric A. McCullough [7] in 2020. His thesis sighted at addressing the proliferation of IoT which again corresponds to botnet attacks using those devices. The process has also managed to bring about a better runtime latency. The main idea revolves around a visual heat map which depicts the ingress and egressing packets over a particular interval and are allowed to be classified by lightweight Convolutional Neural Network (CNN). Segregation of traffic as to which corresponds to DDoS attack and normal requests are provided as a result of classification.

In 2022, Sriram S et al. [9] put forward a solution to the recurrent issue in detection of botnet attacks in IoT devices. The suggested framework worked towards collecting the network traffic and then applied processes to convert them into connection records thereby making it eligible for models to identify threats. Further they have also employed Deep Neural Networks (DNN) to pass labelled connection records for the purpose of identifying important features automatically.

In the work proposed by Mohd Anul Haq et al. [10], the idea was to exploit the Deep Neural Network to detect and classify botnets. Samples from nine compromised industrial

IoT devices were used to feed and train the models that establish and classify common botnet attacks. Effective utilisation of Principal Component Analysis (PCA) turned useful for best extraction of details and in dimensionality reduction that directly affects the classification in such environments. As discussed in [9], GridsearchCV has been employed in designing models with rigorous hyper parameters. An intuitive approach was to use the callback option to pause at different epochs and to prevent overfitting of data. As a result two novel deep neural network-based detection and classification models were obtained tethering with good training and validation accuracy.

Several research papers have been discussed in this section, each contributing to the field of botnet attack detection in IoT systems. However, there are some missing aspects in the existing literature that our paper addresses, along with the some novel contributions. Firstly, previous researchs have overlooked the specific analysis of the IoT Botnet Dataset, which is a distinctive characteristic of our study. By conducting a comprehensive analysis of this dataset, our paper provides insights into its unique characteristics and limitations, contributing to a better understanding of botnet attacks in IoT systems.

Furthermore, while previous works have explored machine learning and deep learning models for botnet attack detection, they have often focused on specific models or datasets. In contrast, our paper investigates the combination of various pre-processing techniques, such as normalization, standardization, PCA, SVD, and SMOTE, with multiple models including KNN, SVM, Random Forest, Decision Tree, ANN, and LSTM. This comprehensive exploration of different models and pre-processing techniques adds new knowledge to the field, enabling a more nuanced understanding of their effectiveness in botnet attack detection.

Additionally, our paper introduces a novel approach to address the issue of bias and imbalance in botnet attack datasets. By combining SMOTE oversampling with K-means clustering, we mitigate the imbalanced nature of the dataset and improve the performance of the models. This novel technique contributes to the existing literature by providing a new method for creating an unbiased training dataset and enhancing the accuracy of botnet attack detection.

In conclusion, our paper bridges the gaps in the previous literature by specifically analyzing the IoT Botnet Dataset, exploring the combination of various pre-processing techniques and models, and introducing a novel approach to address bias and imbalance. These contributions make our research a valuable addition to the field of botnet attack detection in IoT systems.

III. PROPOSED SOLUTION

The Proposed solution as shown in Fig ?? is to firstly convert the biased dataset into an unbiased one using the technique of Smote and also by an unique approach that we perform, by taking the KNN-cluster centroids of the majority rows (in our case rows with target value as non-attack, number

of rows: 3,70,424) and merging it with the minority rows (in our case rows with target value as non-attack, number of rows: 19). This dataset later becomes our training data as we remove the biased nature and make it unbiased, and we use the originally unbiased dataset as our testing data after performing scaling techniques such as normalisation and standardisation on it. To improve the performance of our models we try to implement Dimensionality Reduction techniques such as PCA(principal component Analysis) & SVD(Singular Value Decomposition), where we try to reduce the dimension of our training and testing dataset to improve the performance time of the models. After performing the above said, we would have three sets of training dataset, one after applying PCA, another after applying SVD and another after applying smote. Now we use these three sets of training data on our models one by one and test them with our testing dataset and try to obtain the best combination of the training - testing dataset, Dimensionality reduction technique, and model that gives us the best results with better accuracy.

We are utilising the IoT Botnet Dataset for our analysis. The initial step involves preprocessing the data to prepare it for modelling. We begin by removing duplicate rows. Some columns, such as 'Source_port' and 'destination_port', contain missing values that cannot be logically filled with the mean or mode. Therefore, we choose to fill these missing values with the value '-1'. The target column in the dataset represents a binary classification: '1' indicates an attack, while '0' represents a non-attack. To handle categorical columns, we apply one-hot encoding followed by label encoding, ensuring that all columns are in numeric format.

Next, we perform correlation analysis on the dataset, setting a threshold value of 0.9. We identify columns with correlation coefficients exceeding this threshold and remove one of each highly correlated pair to reduce the number of columns. Subsequently, we apply feature scaling techniques such as standardisation and normalisation to the dataset. Standardisation brings values closer to the mean, while normalisation reduces the scale of values without altering the shape of the distribution curve. We also identify and remove outliers using the interquartile range (IQR) method. By following these data preprocessing steps, we successfully clean the dataset.

To further process the data, we employ dimensionality reduction techniques such as Principal Component Analysis (PCA) and Singular Value Decomposition (SVD). These techniques help reduce the dimensions of the dataset, leading to improved processing time. Given the highly imbalanced nature of the dataset (19 attack rows versus 370,424 non-attack rows), we propose a unique approach. We split the dataset into two parts: one containing only non-attack rows and the other containing attack rows. We then apply K-means clustering to the non-attack rows, creating 100 clusters and obtaining the cluster centroids. Consequently, we represent the 370,424 non-attack rows with these 100 records. Next, we

combine these 100 records with the second part of the dataset, which comprises the 19 attack rows. As a result, the new dataset consists of 119 rows, significantly reducing the bias in the ratio of non-attack to attack rows. This approach enables us to obtain a less biased dataset for training the model, while the original biased dataset obtained after outlier removal is used for testing purposes. Additionally, the Synthetic Minority Oversampling Technique (SMOTE) is applied to the 19 attack rows, increasing them to 100. We merge these 100 attack rows with the 100 records representing the non-attack rows, resulting in a new unbiased dataset with 200 rows.

Having completed the aforementioned steps, we proceed to generate the models. We have three sets of datasets: one with PCA applied, one with SVD applied (both containing 119 rows), and one with SMOTE applied (containing 200 rows). We train the models three times, each time using one of these datasets, in order to identify the combination that yields the best results. The models we train include K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Random Forest, Decision Tree, Artificial Neural Network (ANN), and Long Short-Term Memory (LSTM).

Machine Learning models that are implemented in this research using Scikit-learn and respective Deep Learning models are implemented with the help of Tensorflow and Keras. Categories of classification algorithms were implemented to understand the botnet network characteristics under different circumstances. To extract the details pertaining to detecting the classification between attacks and normal traffic.

The aim of this experiment is to take the extracted data from IOT-Botnet traffic and employ pre-processing techniques to make the data in such a way to use it for different combinations of models that proves to be helpful in detecting botnet attacks [12] [13]. They are expected to aid in the differentiation of data traffic among the models used and support various algorithms either in its raw or manipulated form. This in turn helps in the classification of traffic into attack or normal.

Researches conducted on this category of detection mechanism used an array of models that focussed on increasing the accuracy while retaining the tuning process on modelling part. Although experimental analysis on the modelling part yields a considerable amount of accuracy, it holds true to a certain level and the rest is dependent on the environment it resides in. Hence the performance optimization on the accuracy is discovered to be influenced by major factors like pre-processing and the flexibility of data to be accommodated with different models. The initial consideration was to apply dimensionality reduction techniques to get the data aligned for processing before feeding them into various ML and DL models. Post analysis on null value counts revealed considerable numbers on some of the columns and removed those that were unnamed. Significant features were retained such as "sport" and "dport". Columns corresponding to those of nan values were addressed with respect to rows and replaced with -1.

One-hot encoding method [3] is used to convert categorical information in data into a predefined format that assists in being compatible to the models that it is aimed to feed into. The method is employed to convert the categorical columns into the diverse numerical format. Regular pre-processing method of converting every other data type into integer format inorder for it to be made available to models. Dataset contains some unsupported values which were converted to -1. An important factor influencing the processing steps was the IP address format which was given attention and applied One-hot encoding for the same.

DATASET

The data used for our analysis came from STUDY. It consists of 22 columns. Two are identifiers to distinguish patient and specific trial. Two of the data columns are demographical data, age and sex. Two columns are motor and total UPDRS which represent clinician measurement of symptoms and are considered source of truth. The remaining 16 columns encode jitter, shimmer, and 5 other numeric values which describe characteristics of voice thought to be related to Parkinson's Disease symptom severity.

TABLE I
VOICE SIGNAL PROCESSING FEATURES

Variable Name	Role	Type
Jitter(%)	Feature	Continuous
Jitter(APS)	Feature	Continuous
Jitter:RAP	Feature	Continuous
Jitter:PPQ5	Feature	Continuous
Jitter:DDP	Feature	Continuous
Shimmer	Feature	Continuous
Shimmer(dB)	Feature	Continuous
Shimmer:APQ3	Feature	Continuous
Shimmer:APQ5	Feature	Continuous
Shimmer:APQ11	Feature	Continuous
Shimmer:DDA	Feature	Continuous
NHR	Feature	Continuous
HNR	Feature	Continuous
RPDE	Feature	Continuous
DFA	Feature	Continuous
PPE	Feature	Continuous

EXPERIMENTS AND OBSERVATIONS

The data distribution was assessed through the examination of distribution plots and scatter plots. 2 revealed an evident lack of equal distribution, indicating the presence of outliers and extreme data points. Addressing these outliers was crucial to enhance the performance of the subsequent models and improve anomaly detection. To achieve this, we employed both normalization and standardization techniques, resulting in the transformed dataset depicted in 3. This transformation effectively improved the scalability and distribution uniformity of the data. The removal of outliers was a primary step towards establishing an optimized starting point for parameter fine-tuning and achieving better results. We utilized the IQR method [11] to identify and remove outliers, resulting in a dataset completely free from outliers, while ensuring that the quantile values remained within an acceptable range.

The relationship between the presence of outliers and the dataset after their removal was visually analyzed using box plot visualizations. Furthermore, after careful consideration of correlation factors, we made the decision to eliminate nine non-correlated columns, as illustrated in Fig 4.

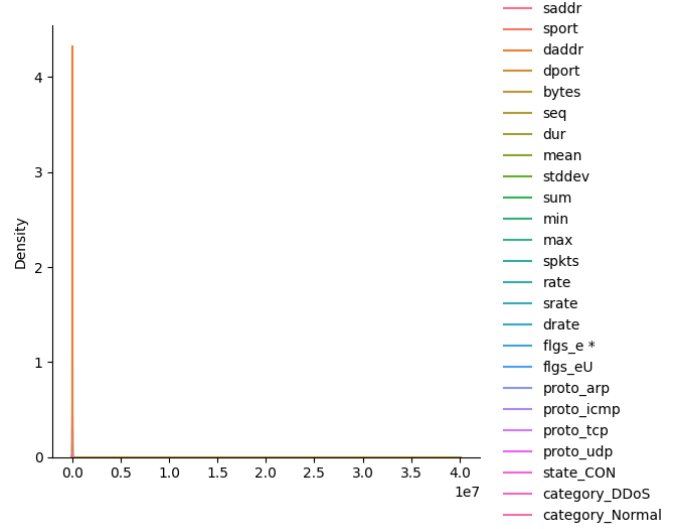


Fig. 2. Visualisation of Dataset before processing

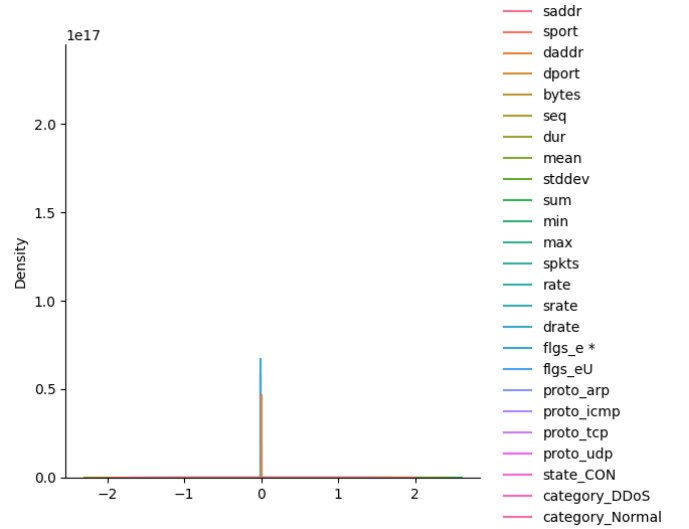


Fig. 3. Visualisation of Dataset post scaling process

Subsequently, the refined dataset was divided into records with attack values of 1, indicating malicious traffic, and records with values of 0, representing non-attack or non-malicious traffic. An exemplary minority was observed in the nonattack traffic category, leading to a significant disparity in the ratio between the two.

Hence the application of synthetic Minority Over-Sampling Technique (SMOTE) [2] was exploited as a solution to have unbiased training . It majorly contributed to the extraction of

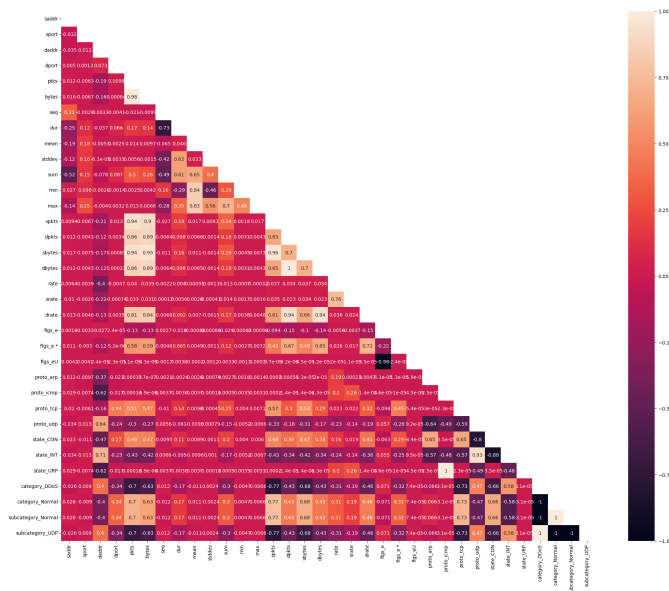


Fig. 4. Correlation factors

minority and oversample it to a desired number. Primarily the goal was to expose data to the following dimensionality reduction mechanisms and apply different combinations of these to multiple ML and DL models to analyse individual results.

- Principal Component Analysis
 - Aided in elimination of 23 columns and resultant data set contained 2 columns
- Singular Value Decomposition
 - Predetermined to scale down the data set to have 2 columns representing the important features after the decomposition

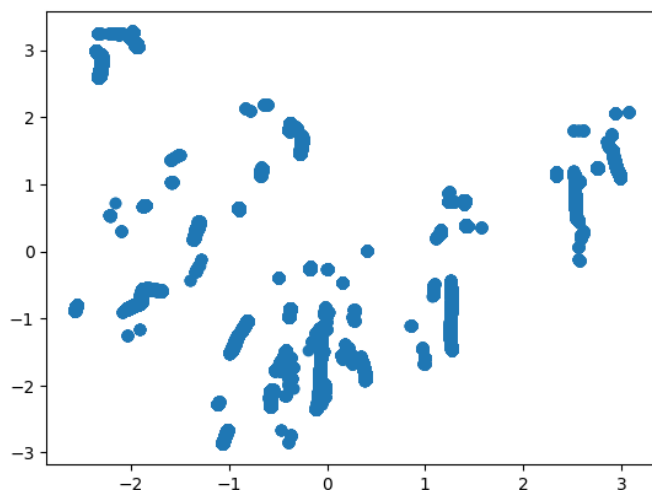


Fig. 5. Datapoints PCA

Moreover, as part of our analysis, we applied a clustering process to the refined dataset in order to identify representative

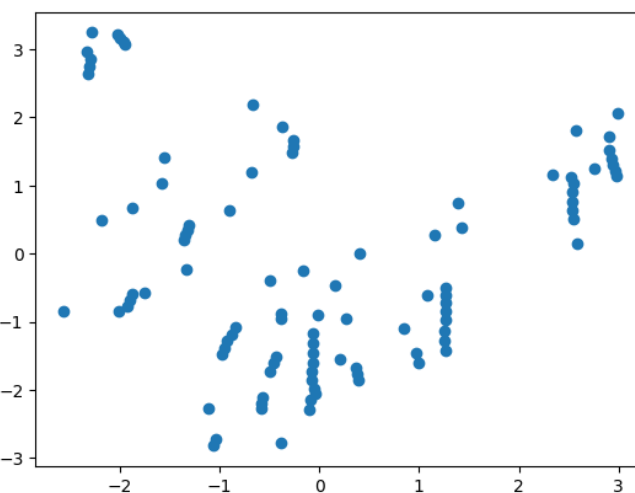


Fig. 6. Cluster Centroids of Datapoints

centroids, as depicted in Fig 5 & Fig 6. To ensure adequate coverage of the entire dataset, we opted for 100 clusters. Our research encompassed various experiments conducted on the processed bot-iot dataset [14], which included:

- Scaling the entire dataset for skew check and outlier removal
 - Normalisation
 - Standardisation
- Majority and Minority split (Class column)
- The Cluster Centroids calculation yielding a new dataset
- Pre-processing methods [PCA,SVD] were carried out on the existing raw dataset
- The Test-Train split up was done to each and every type of data set that was generated as a result of processing
- Data sets were fed into the following models for evaluation:
 - Machine Learning
 - * K-Nearest Neighbour classifier (K-NN)
 - * Support Vector Machine (SVM)
 - * Random Forest (RF)
 - * Decision Tree (DT)
 - Deep Learning
 - * Artificial Neural Network (ANN)
 - * Long-Short Term Memory (LSTM)

Machine Learning models were chosen initially for the metrics evaluation but progressively Deep Learning models came into picture for better evaluation and extensive variety of models to compute and compare accuracy. The performance of these models were determined and brought to compare and contrast each other by the use of metrics such as Accuracy, Precision, Recall, F1-score. Quintessential ML models were trained and tested against different versions of the bot-iot dataset. A duplet of Deep learning models were utilised to observe the runtime and overall performance ratio in comparison

TABLE II
TEST RESULTS FOR CLASSIFYING THE NETWORK FLOW RECORDS AS
ATTACK OR NON-ATTACK WITH PCA PRE-PROCESSED DATA- 0 - 1

Model	Dimensionality Reduction : PCA			
	Precision	Recall	F1-score	Accuracy
KNN	1.0	1.0	1.0	0.9999973
SVM	1.0	0.999954	0.999977	0.9999541
RF (Random Forest)	0.99998	0.99995	0.99996	0.9999406
DT (Decision Tree)	1.0	0.999951	0.999975	0.9999514
ANN	0.999978	1.0	0.999989	0.9652956
LSTM	0.969945	0.99378	0.999972	0.9994868

with ML models, whereas the following are the considerations with respect to the results obtained.

TABLE III
TEST RESULTS FOR CLASSIFYING THE NETWORK FLOW RECORDS AS
ATTACK OR NON-ATTACK WITH SVD PRE-PROCESSED DATA- 0 - 1

Model	Dimensionality Reduction : SVD			
	Precision	Recall	F1-score	Accuracy
KNN	1.0	1.0	1.0	0.9999973
SVM	1.0	0.999991	0.999995	0.9999919
RF (Random Forest)	1.0	0.999962	0.999981	0.9999622
DT (Decision Tree)	1.0	0.999962	0.999981	0.9999622
ANN	1.0	0.999951	0.999975	0.9999514
LSTM	0.969945	0.99378	0.999972	0.9999460

TABLE IV
TEST RESULTS FOR CLASSIFYING THE NETWORK FLOW RECORDS AS
ATTACK OR NON-ATTACK WITH SMOTE PRE-PROCESSED DATA- 0 - 1

Model	Dimensionality Reduction : SMOTE			
	Precision	Recall	F1-score	Accuracy
KNN	0.9999946	1.00	0.9999973	1.00
SVM	1.0	0.999954	0.999977	0.9999514
RF (Random Forest)	1.0	0.999951	0.999975	0.9999514
DT (Decision Tree)	1.0	0.999951	0.999975	0.9999514
ANN	1.0	0.999951	0.999975	0.9999514
LSTM	0.969945	0.999378	0.999972	0.9994868

To accurately identify a generalizable algorithm, the same set of ML classifiers are trained with the same set of parameters on different datasets that are pre-processed with different processes listed above

IV. CONCLUSION

In conclusion, our comprehensive study on analyzing the IoT Botnet Dataset using various machine learning models and pre-processing techniques has yielded valuable insights. However, it is important to acknowledge the limitations of this research. The findings and conclusions may be specific to the characteristics and limitations of the IoT Botnet Dataset and may not directly apply to other datasets or domains. Further research and experimentation on different datasets are necessary to validate the effectiveness of the proposed

approach and ensure its generalizability. Another limitation of this study is the limited feature set used in the analysis. There may be additional relevant features or contextual information that could contribute to improved detection accuracy. Exploring and incorporating such features could enhance the performance of the models and provide a more comprehensive understanding of botnet attack detection.

Additionally, it is important to note that the proposed approach in this study primarily focuses on detecting Distributed Denial of Service (DDoS) attacks. The effectiveness of the models and pre-processing techniques may vary when applied to other types of attacks. This limitation suggests that the approach may not be as applicable or effective in scenarios involving different attack types. Therefore, further research and exploration are needed to develop comprehensive solutions that can address a broader range of attack scenarios in the field of botnet attack detection. Despite these limitations, our research highlights the significance of the choice of pre-processing methods in influencing the performance of machine learning models. Different models demonstrated varying levels of effectiveness depending on the pre-processing techniques employed. Among the evaluated models, K-Nearest Neighbour (KNN) in combination with SMOTE exhibited exceptional performance, achieving the highest accuracy and lowest error rates. This emphasizes the importance of carefully selecting both the model and the pre-processing technique when designing effective solutions for botnet attack detection. Further refinement and validation of the proposed approach on diverse datasets are essential for advancing the field and improving the robustness of botnet attack detection systems.

V. ACKNOWLEDGEMENT

This work is as part of the IBM Shared University Research Grant titled "Detection and Prevention of Advanced Persistent Threat (APT) activities in heterogeneous networks using SIEM and Deep Learning" sponsored by IBM. The authors would like to thank IBM and research lab Vision and Network Analytics (VISHNA) Lab at Amrita School of Computing, Coimbatore for providing the necessary infrastructure for carrying out the work.

REFERENCES

- [1] Alani, M M. (2022, June 30). BotStop : Packet-based Efficient and Explainable IoT Botnet Detection Using Machine Learning - ScienceDirect
- [2] Pokhrel, S., Abbas, R. & Ayal, B. (2021, April 6). IoT Security : Botnet detection in IoT using Machine learning.
- [3] Al-Othman, Z., Alkasasbeh, M., & Baddar, S. (2020). An Efficient Approach to Detect IoT Botnet Attacks Using Machine Learning. Journal of High-Speed Networks. 26. 241–254. 10.3233/JHS-200641.
- [4] Kim, J., Shin, Y., Shim, M., Hong, S., & Choi, E. (2020). Intelligent Detection of IoT Botnets Using Machine Learning and Deep Learning. Applied Sciences. 10.3390/app10197009.
- [5] McDermott, C D., Majdani, F., & Petrovski, A V. (2018) "Botnet Detection in the Internet of Things using Deep Learning Approaches," 2018 International Joint Conference on Neural Networks (IJCNN), Rio de Janeiro, Brazil, 2018, pp. 1-8, doi: 10.1109/IJCNN.2018.8489489.

- [6] Gandhi, R., & Li, Y.(2021) "Comparing Machine Learning and Deep Learning for IoT Botnet Detection," 2021 IEEE International Conference on Smart Computing (SMARTCOMP), Irvine, CA, USA, 2021, pp. 234-239, doi: 10.1109/SMARTCOMP52413.2021.00053.
- [7] McCullough, E A. (2020). Lightweight Deep Learning for Botnet DDoS Detection on IoT Access Networks. MSU Graduate Theses. 3580.
- [8] M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.
- [9] Soe, Y N., Feng, Y., Santosa, P I., Hartanto, R., & Sakurai, K. (2020). Machine Learning-Based IoT-Botnet Attack Detection with Sequential Architecture. Sensors. 20. 4372. 10.3390/s20164372.
- [10] Sriram, S., Vinayakumar, R., Alazab, M., & KP, S.(2020)"Network Flow based IoT Botnet Attack Detection using Deep Learning," IEEE INFOCOM 2020 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), Toronto, ON, Canada, 2020, pp. 189-194, doi: 10.1109/INFOCOMWKSHPS50562.2020.9162668.
- [11] Haq, M A., Khan, M A R. (2021). DNNBoT: Deep Neural Network-Based Botnet Detection and Classification. Computers, Materials and Continua. 71. 10.32604/cmc.2022.020938.
- [12] Benker,K., Gabriel, E.,& Resc, M M.(2008) "Outlier detection in performance data of parallel applications," 2008 IEEE International Symposium on Parallel and Distributed Processing, Miami, FL, USA, 2008, pp. 1-8, doi: 10.1109/IPDPS.2008.4536463.
- [13] Dhanya, K. A., Sulakshan Vajipayajula, Kartik Srinivasan, Anjali Tibrewal, Senthil Kumar, T., & Gireesh Kumar, T. (2023). Detection of Network Attacks using Machine Learning and Deep Learning Models.
- [14] Balaji Bharatwaj, M., Aditya Reddy, M., Senthil Kumar, T.,& Vajipayajula, S. (2022). Detection of DoS and DDoS Attacks Using Hidden Markov Model. In: Ranganathan, G., Fernando, X., Shi, F. (eds) Inventive Communication and Computational Technologies. Lecture Notes in Networks and Systems, vol 311. Springer, Singapore.
- [15] Nair, R., Chugani, M.N.,& Thangavel, S.K.," MetaData: A Tool to Supplement Data Science Education for the First Year Undergraduates", ACM International Conference Proceeding Series,pp.153-160,2020
- [16] Thangavel, S.K., Bkaratki,& P.D., Sankar, A.," Student placement analyzer: A recommendation system using machine learning", ", 4th International Conference on Advanced Computing and Communication Systems, ICACCS 2017,2017
- [17] VISHNA Lab(Vision and Network Analytics Lab), Amrita School of Computing, Amrita School of Engineering, Coimbatore ,link: <https://github.com/vishnalab>