A NOVEL METHOD FOR ANALYSIS OF MALWARE

[1] Vadlampuri Prema Manvi [2] Kaluva Bhavya [3]Bobbala Madhu Bhavana [4]gunda Deepthi

[5] Jaladurgam Ragasai

[1]GUIDE ASSISTANT PROFESSOR [2,3,4,5] UG SCHOLAR

[1,2,3,4,5] RAVINDRA COLLEGE FOR ENGINEERING AND WOMEN

EMAIL: [1]premamanvi95@gmail.com [2]bhavyalucky99@gmail.com

[3]madhubhavana01@gmail.com [4]deepthigunda0608@gmail.com

[5]Jaladurgam.rsai@gmail.com

ABSTRACT

Now a days security has became the key elements of enterprise level application, it is developer responsibility to check with the threats and malicious activities are need to be verified and cross check at each and every level of data transfer. Intruders are trying in many ways to access the application data by spoofing, sniffing and other most popular network based attacks. This proposal implements about analyzing a data set and check with the transaction data processed across server to client. Different machined learning algorithms are implemented in order to check the malware attacks across the network transmission.

I. INTRODUCTION

The breakthrough in internet technology and computer networking have made high speed shared internet possible. The effect of this development is the daily increase in the number of computer systems that have become susceptible to malware attacks. The innovation has made the internet a huge storehouse where resources are virtualized and utilized to the need of users. Despite the immense benefits that the internet revolution has brought, there are numerous challenges that it also poses to the security of computer systems. The conventional computer system is entirely centered on a single host machine running operating system, while several machines connected to the host are running on the guest operating system. The prevalent security threat confronting the users is the attack on a computer system by malicious programs which spread to other computers that have not been infected . The threat posed by malware infections has become a major challenge in the field of computer security over the years. The number of new malware on the internet keep on increasing at an alarming rate even as anti-virus companies are making effort to curtail the trend so as to make the vast number of computer user safe. Malware has evolved over time and is becoming more sophisticated than before. It is now more difficult to detect them. There is therefore the need to invent more efficient techniques that can detect and prevent these attacks.

Malware is a malicious program which infringes on the security of a computer system in terms of privacy, reliability, and accessibility of data. This trend has made academicians and industry practitioners to move from the conventional static detection techniques to more dynamic, sophisticated and spontaneous methods that applies accumulated malware behaviour to detect malware attacks . A malware can simply be defined as a malicious program which the user unsuspectingly install on their machine and later these programs can begin to disrupt the proper operation of the machine or might continue unnoticed and carry out malicious actions without been noticed. When the attacker gains control of the machine, he can then have access to any information stored on the machine. Some of the deceptive approaches used to install malware on the computer system through the internet include repackaging the software, update attack or desire for download. The attacker employs any of the methods mentioned before to create malicious software by inserting a certain type of malware into it before uploading it to the internet. Malware can be described as various types of software which have the capacity to wreak havoc on a computer system or illegally make use of this information without the consent of the users. Malware can be categorized in various types, for instance, Botnet, Backdoor, Ransom ware, Rootkits, Virus, Worms, and Trojan Horse, Spyware, Adware, Scareware and Trapdoor. They are used to attack computer systems and for performing criminal activities such as scam, phishing, service misuse and root access.

II. LITERATURE SURVEY

2.1 Evaluating Machine Learning Classifiers to detect Android Malware

AUTHORS: Prerna Agrawal,Bhushan Trivedi

ABSTRACT: Malware Detection using conventional methods is incompetent to detect new and generic malware. For the investigation of a variety of malware, there were no ready-made machine learning datasets available for malware detection. So we generated our dataset by downloading a variety of malware files from the world's famous malware projects. By performing unstructured data collection from the downloaded APK files and feature mining process the final dataset was generated with 16300 records and a total of 215 features. There was a need to evaluate the performance of the generated dataset with supervised machine learning classifiers. So in this paper, we propose a malware detection approach using different supervised machine learning classifiers. Here supervised algorithms, Feature Reduction Techniques, and Ensembling techniques are used to evaluate the performance of the generated dataset. Machine Learning classifiers are evaluated on the evaluation parameters like AUC, FPR, TPR, Cohen Kappa Score, Precision, and Accuracy. We also represented the results of classifiers using Bar plots of Accuracy and plotting the ROC curve. From the results of machine learning classifiers, the performance of the CatBoost Classifier

is highest with Accuracy 93.15% having a value of ROC curve as 0.91 and Cohen Kappa Score as 81.56%.

2.2 A Static Malware Detection System Using Data Mining Methods

AUTHORS: Usukhbayar Baldangombo, Nyamjav Jambaljav, and Shi-Jinn Horng

ABSTRACT: A serious threat today is malicious executables. It is designed to damage computer system and some of them spread over network without the knowledge of the owner using the system. Two approaches have been derived for it i.e. Signature Based Detection and Heuristic Based Detection. These approaches performed well against known malicious programs but cannot catch the new malicious programs. Different researchers have proposed methods using data mining and machine learning for detecting new malicious programs. The method based on data mining and machine learning has shown good results compared to other approaches. This work presents a static malware detection system using data mining techniques such as Information Gain, Principal component analysis, and three classifiers: SVM, J48, and Na\"ive Bayes. For overcoming the lack of usual anti-virus products, we use methods of static analysis to extract valuable features of Windows PE file. We extract raw features of Windows executables which are PE header information, DLLs, and API functions inside each DLL of Windows PE file. Thereafter, Information Gain, calling frequencies of the raw features are calculated to select valuable subset features, and then Principal Component Analysis is used for dimensionality reduction of the selected features. By adopting the concepts of machine learning and data-mining, we construct a static malware detection system which has a detection rate of 99.6%.

2.3 Mobile Malware Detection using Anomaly Based Machine Learning Classifier Techniques

AUTHORS: A.Hemalatha, Selvabrunda

ABSTRACT: Mobile phones are a significant component of people's life and are progressively engaged in these technologies. Increasing customer numbers encourages the hackers to make malware. In addition, the security of sensitive data is regarded lightly on mobile devices. Based on current approaches, recent malware changes fast and thus become more difficult to detect. In this paper an alternative solution to detect malware using anomaly-based classifier is proposed. Among the variety of machine learning classifiers to classify the latest Android malwares, a novel mixed kernel function incorporated with improved support vector machine is proposed. In processing the categories selected are general information, data content, time and connection information among various network functions. The experimentation is performed on MalGenome dataset. Upon

implementation of proposed mixed kernel SVM method, the obtained results of performance achieved 96.89% of accuracy, which is more effective compared with existing models.

2.4 Credroid: Android Malware Detection By Network Traffic Analysis

AUTHORS: Jyoti Malik and Rishabh Kaushal.

ABSTRACT: Android, one of the most popular open source mobile operating system, is facing a lot of security issues. Being used by users with varying degrees of awareness complicates the problem further. Most of the security problems are due to maliciousness of android applications. The malwares get installed in mobile phones through various popular applications particularly gaming applications or some utility applications from various third party app-stores which are untrustworthy. A common feature of the malware is to access the sensitive information from the mobile device and transfer it to remote servers. For our work, we have confined ourselves to defining maliciousness as leakage of privacy information by Android application. In this paper we have proposed a method named as CREDROID which identifies malicious applications on the basis of their Domain Name Server(DNS) queries as well as the data it transmits to remote server by performing the in-depth analysis of network traffic logs in offline mode. Instead of performing signature based detection which is unable to detect polymorphic malwares, we propose a pattern based detection. Pattern in our work refers to the leakage of sensitive information being sent to the remote server. CREDROID is a semi-automated approach which works on various factors like the remote server where the application is connecting, data being sent and the protocol being used for communication for identifying the trustworthiness (credibility) of the application. In our work, we have observed that 63% of the applications from a standard dataset of malwares are generating network traffic which has been the focus of our work.

2.5 The rise of machine learning for detection and classification of malware: Research developments, trends and challenges

AUTHORS: Daniel Giberta , Carles Mateua , Jordi Planesa

ABSTRACT: The struggle between security analysts and malware developers is a never-ending battle with the complexity of malware changing as quickly as innovation grows. Current state-of-the-art research focus on the development and application of machine learning techniques for malware detection due to its ability to keep pace with malware evolution. This survey aims at providing a systematic and detailed overview of machine learning techniques for malware detection and in particular, deep learning techniques. The main contributions of the paper are: (1) it provides a complete description of the methods and features in a traditional machine learning workflow for malware

detection and classification, (2) it explores the challenges and limitations of traditional machine learning and (3) it analyzes recent trends and developments in the field with special emphasis on deep learning approaches. Furthermore, (4) it presents the research issues and unsolved challenges of the state-of-the-art techniques and (5) it discusses the new directions of research. The survey helps researchers to have an understanding of the malware detection field and of the new developments and directions of research explored by the scientific community to tackle the problem.

III. SYSTEM ANALYSIS

3.1 OBJECTIVE OF THE PROJECT

In order to test the application a malware file is uploaded to the system and applying machine learning algorithm to check the malware file contains any signature that effects the data or files in the system.

This malware are highly affecting the files transmitted across distributed environments. Each malware type has different characteristics and a specific feature to identify them based on the parameters mentioned in the trained datasets

After classification results show the distribution of higher effective malware files across various types of networks in the real world.

3.2 THE EXISTING SYSTEM

Network security is the key challenge for many enterprise level application which are mainly used identify the security threats associated across different types of attacks by hackers as well as intruders who always involve in spoofing the data during the data transmission.

The following report has been prepared in light of the analysis of the various problems associated with network and cyber security based datasets and the applications which are using this datasets across various enterprise level architectures like network and cyber security, This methodology helps in Building Development, k-means analysis and Machine Learning Implementations. It has been aptly demonstrated that the effective integration of Data Analytics methodologies has led to economized efforts in the practices of the various organizations facilitating their operations by drawing crucial insights from the relevant data in light of the concealed patterns, preferences of Consumers, trends prevalent in the market and the respective unknown correlation of the different elements under various circumstances of the environment.

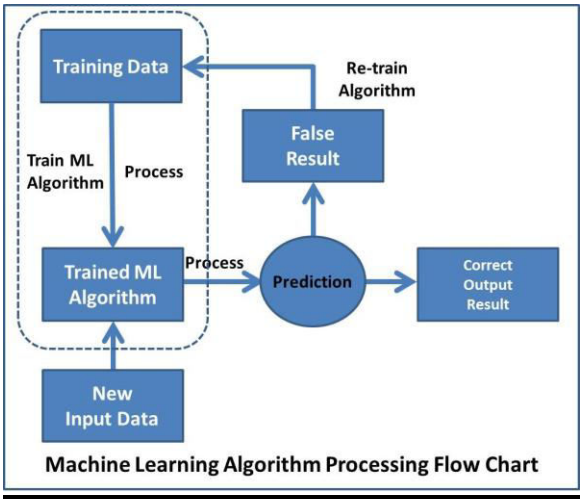
3.3 THE PROPOSED SYSTEM

Implementing comparative evolution of SVM, KNN, ANN-BAT, ANN-PSO with different files with malware are detected and classified by existing malware files in the trained datasets. These classification techniques are more advanced and high accurate in predicting the results based on different data elements provides. Classification helps in finding the feasible

solution in identifying the best matching methods to the given data and results concludes and proves that identification of malware files across the distributed environment, these techniques can also identify in continuous data like smtp, ftp servers and cloud computing environment. Here mentioned a comparative table across KNN,SVM, BR-ANN in best accuracy prediction techniques using machine learning.

4.SYSTEM DESIGN

4.1 SYSTEM ARCHITECTURE:



Machine Learning Algorithm Processing Flow Chart

IV. IMPLEMENTATION:

4.1.Data Collection

The dataset used in this paper is from cooperative bank . This step was done by the original owners of the dataset. And the composition of the dataset**.**understand the relationship among different features. A plot of the core features and the entire dataset.The dataset is further split into 2/3 for training and 1/3 for testing the algorithms. Furthermore, in order to obtain a representative sample, each class in the full dataset is represented in about the right proportion in both the training and testing datasets. The various proportions of the training and testing datasets used in the paper.

4.2.Data Preprocessing

The data which was collected might contain missing values that may lead to inconsistency. To gain better results data need to be preprocessed so as to improve the efficiency o the algorithm. The outliers have to be removed and also variable conversion need to be done. In order to overcoming these issues we use map function.

4.3.Model Selection

Machine learning is about predicting and recognizing patterns and generate suitable results after understanding them. ML algorithms study patterns in data and learn from them. An ML model will learn and improve on each attempt. To gauge the effectiveness of a model, it's vital to split the data into training and test sets first. So before training our models, we split the data into Training set which was 70% of the whole dataset and Test set which was the remaining 30%. Then it was important to implement a selection of performance metrics to the predictions made by our model. In this case, we tried to identify

whether an individual is going to default on a stock market.

4.4 Predict Malware

The total number of features within malware file and then convert it into grey colour image and then apply PSO to extract features and then apply ML algorithm to predict whether uploaded file contains malware signature or not

V. MODULE DESCRIPTION

To implement this project we will design following modules

1) Upload dataset: using this module we will upload malware dataset to application
2) Preprocessing or data exploratory method: using this module we will read binary values from dataset and then convert it into grey colour images and then explore count of each attack available in dataset.
3) Feature selection/optimization: for better prediction and to reduce features we are applying PSO feature selection algorithm which will select 600 relevant features from available 1000 features in each image.
4) Run SVM Algorithm: Using this module we will split dataset into train and test and then build SVM trained model. Trained model will be applied on test data to calculate and test prediction accuracy
5) Run KNN Algorithm: Using this module we will split dataset into train and test and then build KNN trained model. Trained model will be applied on test data to calculate and test prediction accuracy
6) Run ANN Algorithm: Using this module we will split dataset into train and test and then build ANN trained model. Trained model will be applied on test data to calculate and test prediction accuracy
7) Comparison Graph: Using this module we will plot various metrics graphs such as accuracy, precision, recall and fmeasure to evaluate performance of all algorithms.
8) Predict Malware from New File: Using this module we will upload new malware file and then convert it into grey colour image and then apply PSO to extract features and then apply ML algorithm to predict whether uploaded file contains malware signature or not

VII. Classification Algorithms

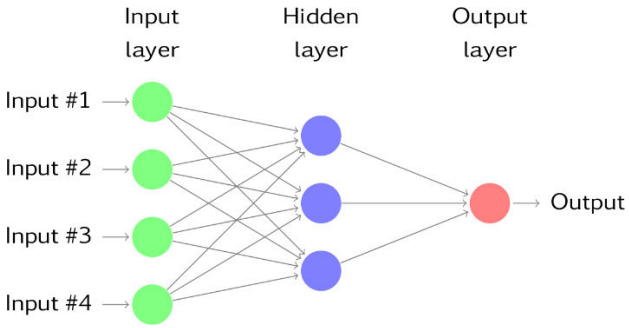
Classification algorithms work by predicting the best group to which a data point belongs to by learning from labeled observations. It uses a set of input features for the learning process. Classification algorithms are good for grouping data that are never seen before into their various groupings and are therefore extensively used in machine learning tasks. Some of the well-known classification algorithms used in this paper are briefly discussed below:

6.1ANN algorithms Details

To demonstrate how to build a ANN neural network based image classifier, we shall build a 6 layer neural network that will identify and separate one image from other. This network that we shall build is a very small network that we can run on a CPU as well. Traditional neural networks that are very good at doing image classification have many more parameters and take a lot of time if trained on normal CPU. However, our objective is to show how to build a real-world convolutional neural network using TENSORFLOW.

Neural Networks are essentially mathematical models to solve an optimization problem. They are made of neurons, the basic computation unit of neural networks. A neuron takes an input (say x), do some computation on it (say: multiply it with a variable w and adds another variable b) to produce a value (say; z= wx+b). This value is passed to a non-linear function called activation function (f) to produce the final output(activation) of a neuron. There are many kinds of activation functions. One of the popular activation function is Sigmoid. The neuron which uses sigmoid function as an activation function will be called sigmoid neuron. Depending on the activation functions, neurons are named and there are many kinds of them like RELU, TanH.

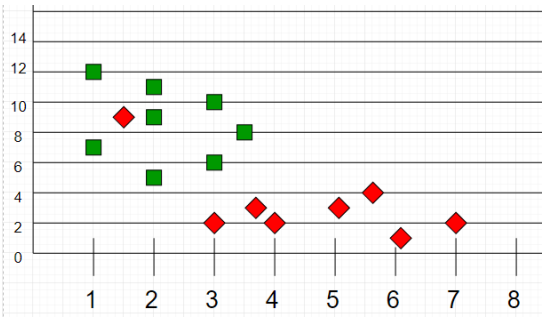
If you stack neurons in a single line, it's called a layer; which is the next building block of neural networks. See below image with layers



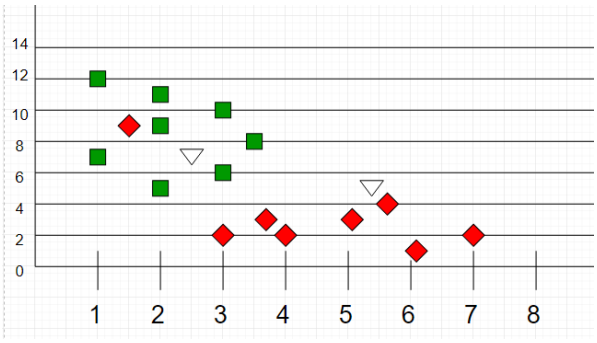
To predict class label multiple layers operate on each other to get best match layer and this process continues till no more improvement left.

6.2 K NEAREST NEIGHBOR ALGORITHM

K-Nearest Neighbors is one of the most basic yet essential classification algorithms in Machine Learning. It belongs to the supervised learning domain and finds intense application in pattern recognition, data mining and intrusion detection. It is widely disposable in real-life scenarios since it is non-parametric, meaning, it does not make any underlying assumptions about the distribution of data (as opposed to other algorithms such as GMM, which assume a Gaussian distribution of the given data). We are given some prior data (also called training data), which classifies coordinates into groups identified by an attribute. As an example, consider the following table of data points containing two features:



Now, given another set of data points (also called testing data), allocate these points a group by analyzing the training set. Note that the unclassified points are marked as 'White'.

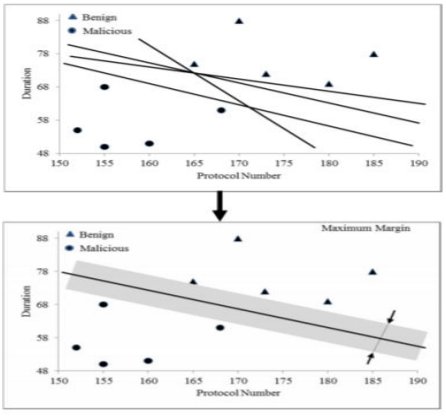


SUPPORT VECTOR MACHINE(SVM)

"Support Vector Machine" (SVM) is a supervised machine learning algorithm which can be used for both classification or regression challenges. However, it is mostly used in classification problems. In this algorithm, we plot each data item as a point in n-dimensional space (where n is number of features you have) with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the hyper-plane that differentiate the two classes very well (look at the below snapshot). The SVM algorithm is implemented in practice using a kernel. The learning of the hyperplane in linear SVM is done by transforming the problem using some linear algebra, which is out of the scope of this introduction to SVM. A powerful insight is that the linear SVM can be rephrased using the inner product of any two given observations, rather than the observations themselves. The inner product between two vectors is the sum of the multiplication of each pair of input values. For example, the inner product of the vectors and 2*5 + 3*6 or 28. The equation for making a prediction for a new input using the dot product between the input (x) and each support vector (xi) is calculated as follows:

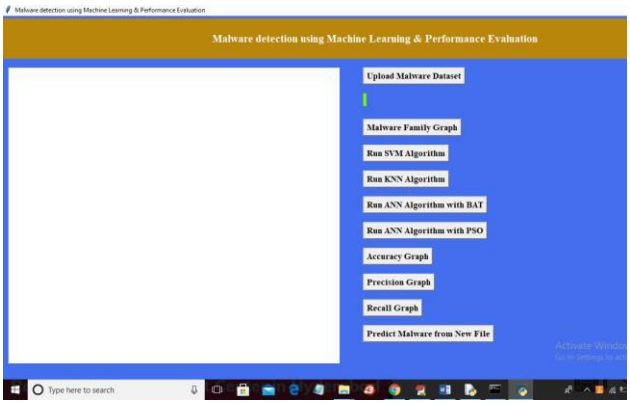
$$f(x) = B0 + sum(ai * (x,xi))$$

This is an equation that involves calculating the inner products of a new input vector (x) with all support vectors in training data. The coefficients B0 and ai (for each input) must be estimated from the training data by the learning algorithm.

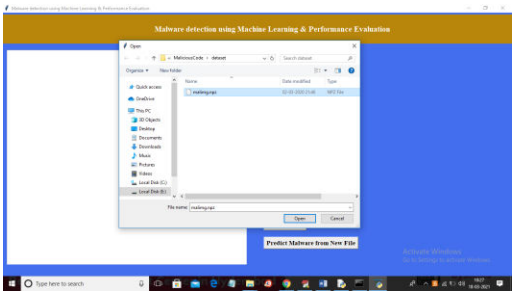


7.SCREENSHOTS

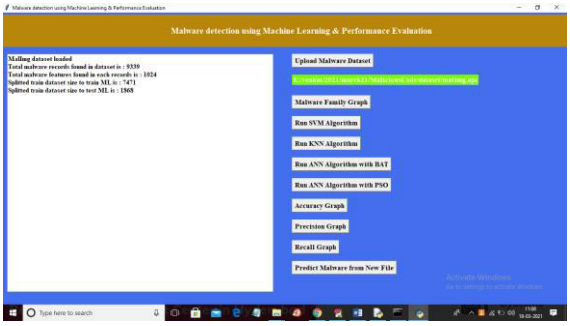
To run project double click on 'run.bat' file to get below screen



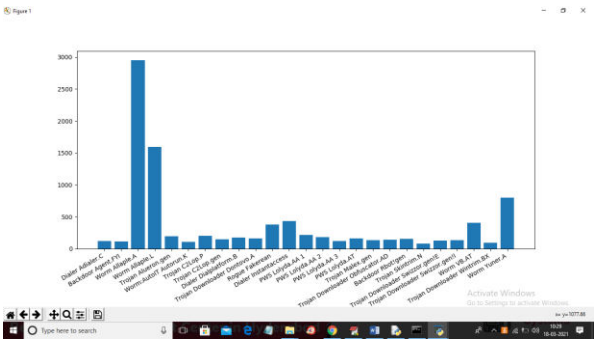
In above screen click on 'Upload Malware Dataset' button and upload dataset



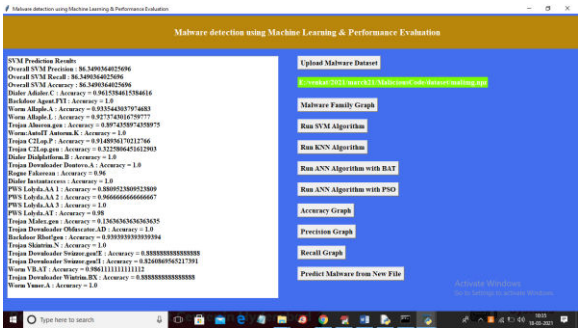
In above screen selecting and uploading 'malimz.npg' malware dataset file and then click on 'Open' button to load dataset and to get below screen



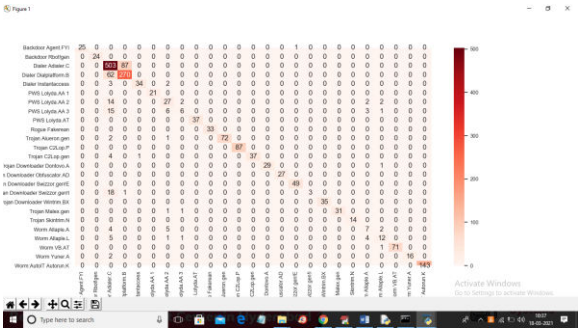
In above screen dataset loaded and we can see dataset contains total 9339 records from different malware attacks and each attack contains 1024 features and then application splitting dataset into train and test where 7471 records are using for training and 1868 records are using for testing and now click on 'Malware Family Graph' button to get below graph



In above graph x-axis represents malware family name and y-axis represents count of that attack available in dataset and now close above graph and then click on 'Run SVM Algorithm' button to train above dataset with SVM

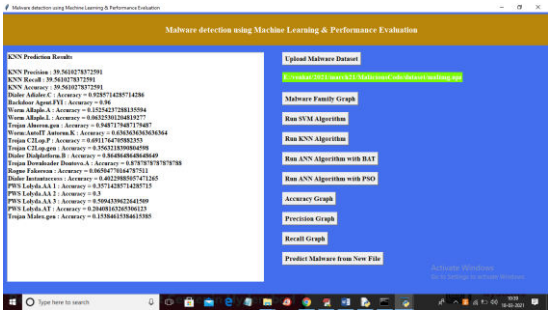


In above screen we can see SVM final accuracy, precision and recall in first 3 lines and then we calculate accuracy based on each attack family.

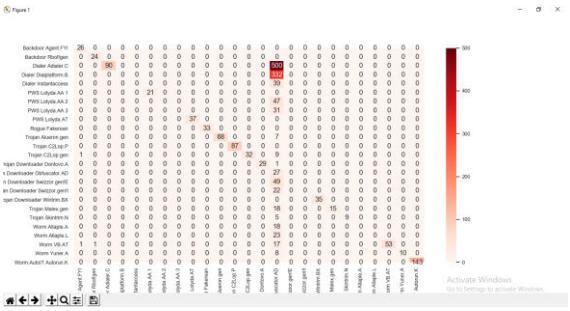


In above graph we can see confusion matrix from each attack prediction and in above graph x and y-axis represents attack name and all numeric values contains count of that attack prediction

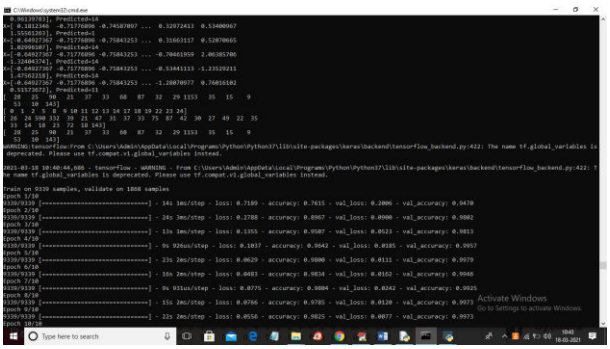
Now click on 'Run KNN Algorithm' button to build KNN modeland to get below screen



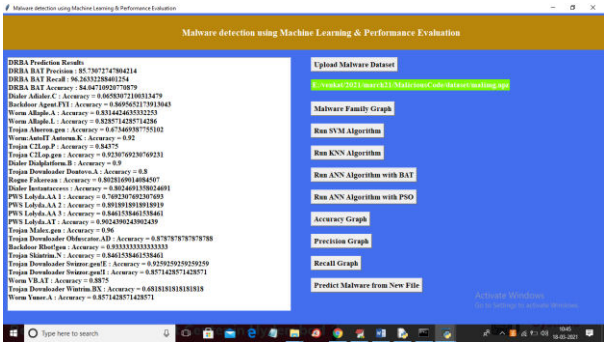
In above screen we can see KNN final accuracy and accuracy of each attack class and below is the confusion matrix graph for KNN



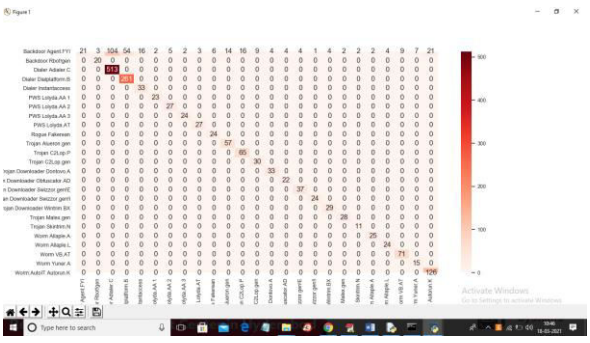
Now click on 'Run ANN Algorithm with BAT' button to optimize dataset features with BAT and then build ANN model and calculate prediction accuracy



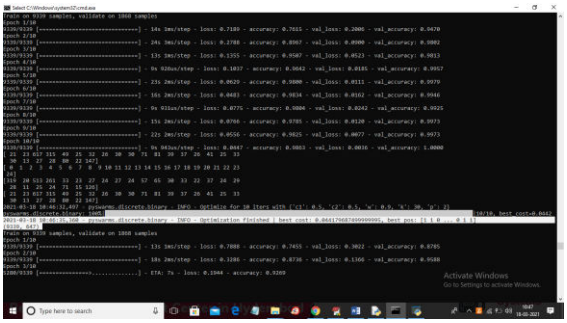
In above graph for ANN we took 10 epoch and at each increasing epoch accuracy get increase and loss get decrease which indicates features optimize successfully and model is trained with more than 80% accuracy



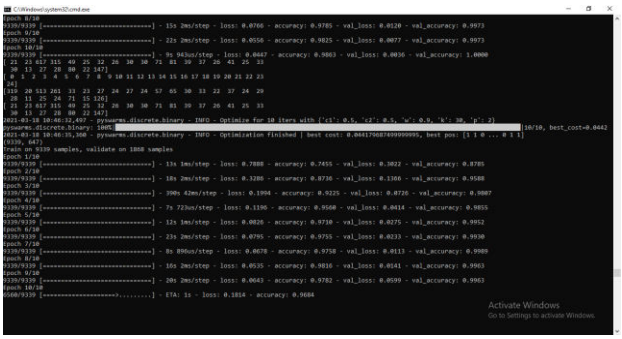
In above screen we can see ANN BAT final accuracy, precision and recall and then got accuracy of each attack class and below is the ANN confusion matrix graph for each attack prediction count



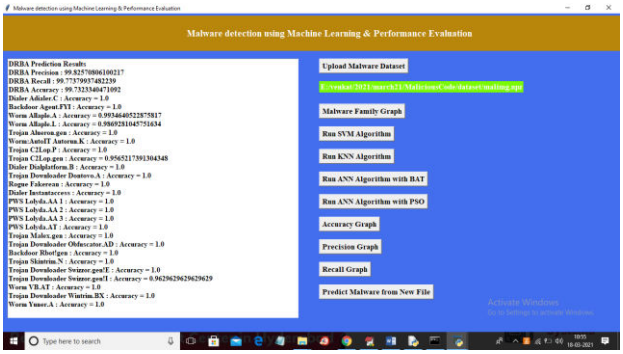
Now click on 'Run ANN with PSO Algorithm' button to optimize features with PSO



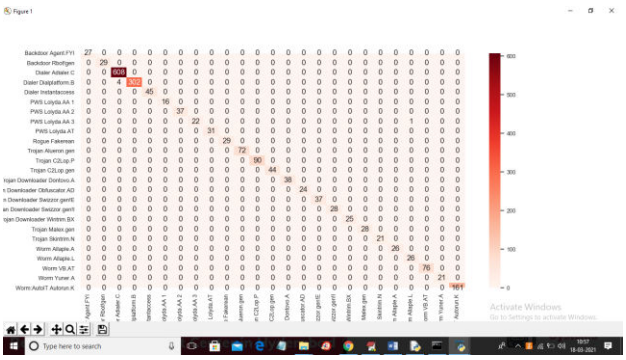
In above screen we can see PSO optimization started and we are using PSO algorithm from PYSWARM package and in selected text we can see features reduce from 1000 to 647 where 9339 are the total records and 1000 are the features in each record



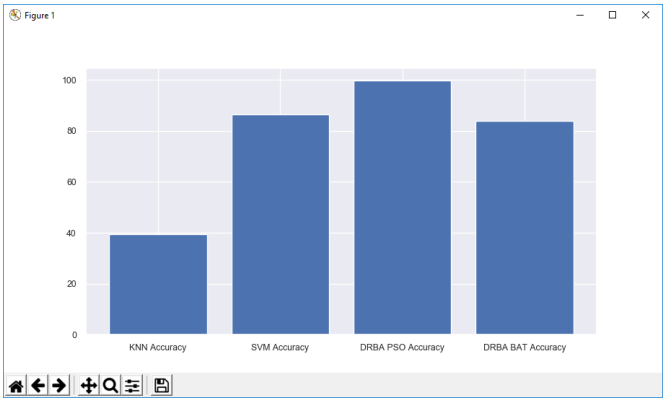
In above screen for PSO ANN also we took 10 iterations/epoch and at each epoch accuracy is getting better and loss getting reduce



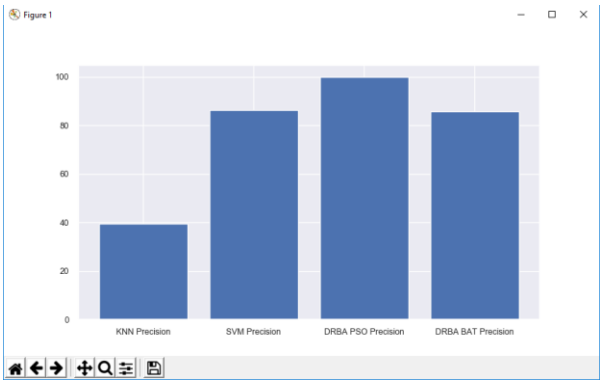
In above screen ANN with PSO got more than 99% precision, recall and accuracy and for almost all family names we got accuracy prediction as 1.0 which means 100% accurate prediction and below is the confusion matrix graph



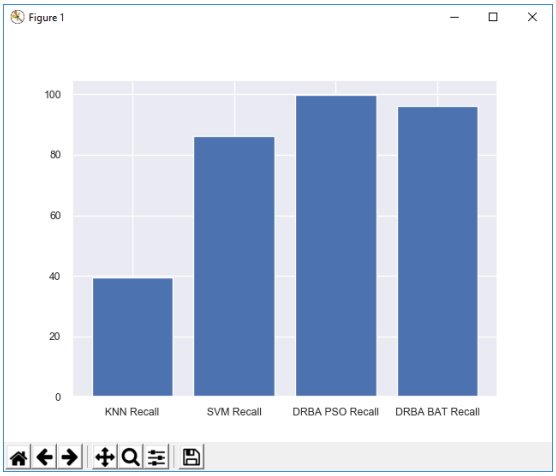
Now click on 'Accuracy Graph' to get below graph



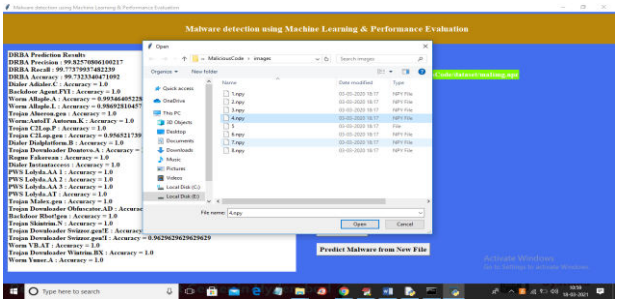
In above screen PSO with ANN got high performance and in above graph x-axis represents algorithm name and y-axisrepresents accuracy. Now click on 'Precision Graph'



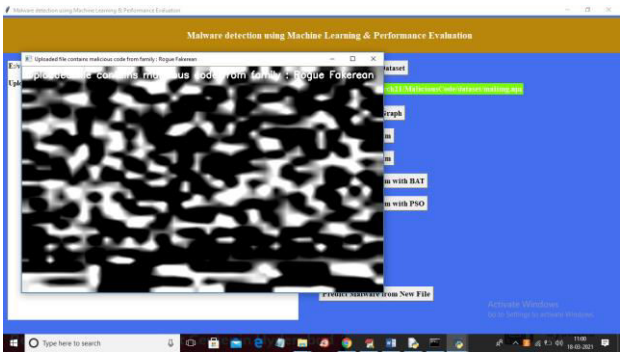
Now click on 'Recall Graph'



Now click on 'Predict Malware from New File' button to upload malware file and get predicted family name



In above screen selecting and uploading 4.npy file and then click on 'Open' button to get below screen



In above screen uploaded file first convert into grey colour image and then apply ANN model on that image to get prediction result detected malware family and you can see predicted name in image title bar or white colour text. Similarly you can upload other files and test it.

8.CONCLUSION

This implementation mainly highlights identification of malware files during the data transmission, these malware files changes the order of packets and leads mainly practical issues like packets lose, ipspoofing and other D-DOS attakcs. By using this project by implementing various machine learning algorithms and used to classify the malware files comparing with the existing malware family and identifying whether the given files match with the appropriate properties of the given malware family files and classifying them and predicting the malware type of files with several test data.

FUTURE SCOPE

The limitations and the future scope of the Data as mentioned above Analytics technologies in light of optimal decision undertaking within organizational context while considering the various risk

management approaches fostering enhanced consumer experience and improved innovation and development practices within industries have been duly considered.

9. REFERENCES

[1]. Sanjay Chakrabortya and Lopamudra Dey. A rule-based probabilistic technique for malware code detection. Multiagent and Grid Systems – An International Journal, IOS Press, 12, 2016, pp. 271–286 271. DOI 10.3233/MGS-160254

[2]. Y. Zhou, Z. Wang, W. Zhou, and X. Jiang. Hey, you, get off of my market: Detecting malicious apps in official and alternative android markets. in NDSS, vol. 25, no. 4, 2012, pp. 50–52.

[3]. D. Keragala. Detecting malware and sandbox evasion techniques, SANS Institute InfoSec Reading Room, 2016. URL: https://www.sans.org/reading-room/whitepapers/ forensics/detecting-malware-sandbox-evasion-techniques-36667.

[4]. Sharif, M., Yegneswaran, V., Saidi, H., Porras, P., and Lee, W. Eureka: A framework for enabling static malware analysis. In Computer security-ESORICS 2008, pages 481- 500. Springer.

[5]. Moser, A., Kruegel, C., and Kirda, E. Limits of static analysis for malware detection. In Computer security applications conference, ACSAC 2007. Twenty-third annual, 2007, pages 421-430.

[6]. Egele, M., Scholte, T., Kirda, E., and Kruegel, C. A survey on automated dynamic malware-analysis techniques and tools. ACM Computing Surveys (CSUR), 2012, 44(2):6.

[7]. Ahmad, S., Ahmad, S., Xu, S., and Li, B. Next generation malware analysis techniques and tools. In Electronics, Information Technology and Intellectualization: Proceedings of the International Conference EITI 2014, Shenzhen, 16-17 August 2015, page 17. CRC Press.

[8]. Gorecki, C., Freiling, F. C., Kuhrer, M., and Holz, T. Trumanbox: Improving dynamic malware analysis by emulating the internet. In Stabilization, Safety, and Security of Distributed Systems, Springer, 2011, pages 208-222.

[9]. Jyoti Malik and Rishabh Kaushal. Credroid: Android Malware Detection By Network Traffic Analysis, ACM PAMCO'16, July 05 2016, Paderborn, Germany, DOI: http://dx.doi.org/10.1145/2940343.2940348

[10]. L. Tenenboim-Chekina, O. Barad, A. Shabtai, D. Mimran, L.Rokach, B. Shapira and Y. Elovici. Detecting Application Update Attack on Mobile Devices through Network Features. In INFOCOM 2013.

[11]. Mahinthan Chandramohan and Hee Beng Kuan Tan. Detection of Mobile Malware in the Wild. In Computer, ieeexplore.ieee.org, (Sept. 2012) vol. 45, pp. 65-71.

[12]. Mohamad Baset. Machine Learning for Malware Detection. MSc. Dissertation, School of Mathematical and Computer Sciences, Heriot-Watt University, 62 pages, 2016.

[13]. T. G. M. Van Erp, T. D. Cannon, H. L. Tran, A. D. Wobbekind, M. Huttunen, J. Lonnqvist, J. Kaprio, O. Salonen, L. Valanne, V. P. Poutanen, C. G.

Standertskjold-Nordenstam, A. W. Toga, and P. M. Thompson. Genetic influences on human brain morphology, in IEEE International Symposium on Biomedical Imaging: Nano to Macro, 2004., April 2004, Vol. 1, pp. 583–586

[14]. Cisco (2015). What is the difference: Viruses, worms, trojans, and bots? Online. http://www.cisco.com/web/about/security/intelligence/virus-worm-diffs.html.

[15]. Zeidanloo H. R., Tabatabaei S. F., Amoli P. V. and Tajpour A. All About Malwares (Malicious Codes), 2015.