International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

COPY RIGHT

ELSEVIER SSRN

IJIEMR Transactions, online available on 26th Apr 2022. Link
:http://www.ijiemr.org/downloads.php?vol=Volume-11&issue= Spl Issue 04

DOI: 10.48047/IJIEMR/V11/SPL ISSUE 04/03

Title **DETECTION OF ANDROID MALWARE USING GENETIC ALGORITHM BASED ON OPTIMIZED FEATURE SELECTION**

Volume 11, SPL ISSUE 04, Pages: 22-39

Paper Authors
Mr. Anand Thota ,Shaik.Raheem, P. Krishna Mouli, P.Bhargav,B.Lakshmi Prasad

USE THIS BARCODE TO ACCESS YOUR ONLINE PAPER

To Secure Your Paper As Per UGC Guidelines We Are Providing A Electronic Bar Code

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

DETECTION OF ANDROID MALWARE USING GENETIC ALGORITHM BASED ON OPTIMIZED FEATURE SELECTION

Mr. Anand Thota [1],Shaik.Raheem[2], P. Krishna Mouli[3], P.Bhargav[4],B.Lakshmi Prasad[5].

[1]Associate professor, CSE, PSCMR College of Engineering & Technology, Vijayawada,Andhra Pradesh,

[2,3,4,5]Students, CSE, PSCMRCET, Vijayawada, Andhra Pradesh

anand4uammu@gmail.com[1],raheemlito786@gmail.com[2],siddhunaidur@gmail.com[3],peyyalabhargav214@gmail.com[4],lakshmiprasadbandari9177@gmail.com[5]

ABSTACT:

Android is a open source operating system which is developed by google.Most of the people use android mobiles.So they have become a major target for the cyber attackers.This study focuses on machine learning approach for malware detection which uses evolving Genetic algorithm.Genetic algorithm is helpful in achieving optimization.machine learning classifiers are trained with important features from the genetic algorithm,they can be able to detect malware is evaluated before and after feature selection.Researchs tell us that Genetic algorithm gives the best feature set,with feature dimension reduced to less than half .By this we can achieve an accuracy of 94 percent.

KEYWORDS:

AndroidMalwareDetection(AMD),GeneticAlgorithm(GA),MachineLearning(ML),SupportVectorMachine(SVM),Neural Networks(NN),Optimized Feature Selection using Genetic Algorithm (GA).

1.INTRODUCTION

Playstore is the one and only app store that is majorly used by android users to download android apps.Because of the its freely available nature,more attackers are targeting on android operating system.So attackers developing malicious applications that can infiltrate the devices and can grab the sensitive data from the victim or user.And they can blackmail them and they can do more harmful threats Sensitive information emails,phonebook contacts,and information.These information can be mointered remotely from a different location.These malicious apps can pose a major threat to the smartphone users.So malware analysis is needed for detecting harmful application .There are two types of attacks static and dynamic analysis.We require signature database rather than signature-based method so that it is updated on regular basis.

2.RELATED WORK

.In [1],According to Halil Murat Unver et al., This is rise in the Smartphone usage and Android is most popular Operating system. These malware is found in every smartphone. That results malware writers had turned eyes on Android. This results shows a proposing a malware classified methodology for detecting malware in the Android. A view based process is presented for finding android applications as a benign ware or Malware. In this model, we are converting the contents of a particular application into grayscale images and those images are classified in the image processing techniques and machine learning algorithms. In this model we are taking three grayscale image datasets. In the first data set, The application is opened and manifest decompiled.XML files are used to made a grayscale images. In the second dataset, the grayscale images are translated into DEX code. The three components (Manifest, DEX and resources) are deployed into third dataset. Every application ARSC files are translated into grayscale images The features all are combined into a single feature using BOVW method. If we evaluated this AMD dataset . It contains nearly 24000 samples from 71 families.The accuracy will gives better than 98 percent.

In [2], Hyoil Han et.al. stated that nearly 65% people who were using smart phones

using are in dangerous situation. Humans using their login id and passwords for using their favourite applications. Some of them are malicious and vulnerable to users. For solving that issue applications using an api interface to using the resources.In this model, we are performing Support Vector Machine that performs already existed approachesfor finding or detecting the malicious Android APK. Each API call can be represented by triple parameters(API method name, argument and return type) those are invoking by an android APK. SVM can easily classified into different categories such sparse high dimensional data. In this model, the accuracy is occurring more than 90 %.

In [3], Dong ji Wu et.al. They proposed an analysis for detecting a Android Malware. we have different components to measure android application Behaviour. Those components are intent messages, deployment and API requests. For detecting malware developed a mechanism is known as "Droid Mat ".The data is retrieved by Droid mat. We performed different clustering techniques to finding a different types of Malware. Droid mat converts the application into a fully automation. Misuse Detection is used to detect a malware based on signature method. In these model we are using clustering mechanisms which is divided into three categories, dynamic, static and Extra information. We have 238 varities of malware,34 families and 1500 good apps. Finally, we used the KNN algorithm to solve the harmful malwares. Droid mat is more effective than Androguard. It gives 97.87% accuracy

In [4],Hyo-Sik Ham and his associates As a result of the rapid increase in the number of users, as well as its numerous weaknesses, Android malware is on the rise. The suggested system's structure for identifying malware and monitoring features is built using malware detection architecture and feature definition. Malware detection and analysis components make up the majority of Android malware detection. The data is first processed by a resource collector, which maintains track of the various resources used when a user runs an app through the Android device's agent. Each app's monitored resource data is confirmed and saved to the device. After that, the malware is discovered and assessed. It

analyses each classifier's detection performance by vectorizing the resource data. For Each app is checked before being installed on the device. After that, the malware is discovered and assessed. It analyses each classifier's detection performance by vectorizing each app's resource data and applying four methods of machine learning. Classifiers include naive bayesian, random forest, logistic regression, and SVM-support vector machine. Finally, to detect malware on an Android smartphone, we use network, SMS, CPU, and power categories, which track the rate at which allotted resources are utilised while the programme is running. According to the trial's findings, the Random Forest classifier fared the best in terms of TPR/FPR. Although the SVM classifier underperformed the Random Forest classifier in terms of TP (True Positive) of normal type data and FP (False Positive) of FP (False Positive) of FP (False Positive) of FP (False Positive) of FP (F (False Positive)

In [5], Justin Sahs and his coworkers had a lot of fun at work. They extract functionality from packaged Android apps using Androguard, an open-source software (APKs). We next train a One Class SVM with these extracted features using the Scikit-learn framework, which provides a straightforward interface to LIBSVM. The main idea is to create a classifier that will classify the majority of the training data as positive and only classify training or testing data as negative if it differs significantly from the training data, which is ideal for our purposes because benign Android applications are far more widely available than malicious Android applications, which is ideal for our purposes because benign Android applications are far more widely available than malicious Android applications. The One-Class SVM is a support vector machine that solves a constrained quadratic optimization problem to produce a linear classifier in a high-dimensional feature space.

Table:1 Existing System

S.NO	AUTHOR NAME	ALGORITHM	ADVANTAGES	DISADVANTAGES
1	Halil Murat Unver	Random Forest	It does not suffer from the overfitting problem. Accurate, efficiency and eliable algorithm	Better-Performance is obtained with more Parameters for various algorithms
2	Hyoil Han	SupportVector Machine(SVM) algorithm	Suitable for high dimensional data,Better-Performance,Improved accuracy.	High-computational complexity
3	Dong ji Wu	KNN Algorithm	More accurate, cost efficient. New data can be added seamlessly	perform better and domination of distances by irrelevant attributes
4	Hyo-Sik Ham	Naive Bayes	Gives better results for relatively larger datasets, works quickly and can save a lot of time	Implicitly assumes that all the attributes are mutually independent and sometimes result zero frequency
5	J. Sahs	SupportVector Machine(SVM)	Suitable for high dimensional data,Better-Performance,Improved accuracy.	High-computational complexity

3.PROPOSED SYSTEM

Utility and entertainment apps are the two types of app categories. Permissions play a major role in accepting or denying the malicious apps into the mobile. Other components like Activity ,Services and Content providers are extracted through reverse engineering. These properties are used as feature vector in CSV format. The labelling is done for goodware and malware as 1 and 0 respectively. The CSV fed into a Genetic algorithm which selects the most efficient set of features. Here we used two Machine learning classifiers, Those are Support Vector Machine(SVM) and Neural Networks(NN).The suggested methods collecting a static features from AndroidManifest.xml. In that xml file contains the information of apps with a supporting Android Platform.

A. Pre-processing

All the previous studies have proven that, data cleaning and dimensionality reduction have its impact on the efficiency of ML and deep learning systems. In the process of data cleaning, the proposed system has identified that approximately 3512 entries have missing data and it uses the traditional imputation technique, since most of the attributes don't exhibit the skewness property all the absent values are replaced with their respective present values. The next important factor in the pre-processing is data transformation. The numerical values have a vital role in the computation time rather than the categorical data and more than all the attributes in the dataset have few unique values in their respective columns. So, the system has applied a label encoding technique to convert the unqualified data into numerical data. The label encoding assigns a number to each unique value present for the attribute. Suppose, an attribute known as "chorography" has 4 unique values transformation is also a key pre-processing step, which is applied as standard feature scaling in the proposed system, which is calculated as the "difference between the attribute and the whole mean is divided by the standard deviation". This standard scaler mechanism makes the mean value to be concentrated around 0 and standard deviation values around 1, which in turn helps the data to be in perfect normalization distribution form.

The proposed system uses neural networks to classify the data about the telecom users

and to achieve good accuracy, different optimizers are applied. The step size calculation may affect the magnitude of the features. The major intention of scaling is not to get a bias effect because of the features with high magnitudes.

4.Feature selection

A. Genetic Algorithm

The Genetic Algorithm is used to detect malware by training selected features to machine learning classifiers and evaluating the detection capacity before and after feature selection. This experiment revealed that the genetic algorithm produces the most optimized feature output and assists in cutting the feature dimension in half. 2.4.2.2 Genetic Algorithm for Feature Selection The genetic algorithm is a natural selection-based optimization tool. In this post, I demonstrate how to choose features using genetic algorithms. While scikit-learn has a number of well-known feature selection algorithms, feature selection goes far beyond what is provided there. A fundamental part of any machine learning pipeline is feature selection. However, there is an abundance of data available these days. As a result, there is frequently an excess of features.

Numerous features are redundant, as is often the case with many features. They add noise to your model, making it difficult to comprehend. The issue is determining which characteristics are relevant to the issue. The goal is to have high-end features. This package is compatible with existing sklearn models and offers a lot of genetic selection features and options. For feature selection in this post, I'm utilizing a genetic algorithm. A genetic algorithm, on the other hand, can be used to optimize hyperparameters. Because the stages are simple and broad, they can be used to a variety of situations.

5. Classification

Classification is a mechanism for dividing a data into a combination of categories. The fundamental purpose of a classification challenge is to decide which category or class new data belongs to. A classifier is an algorithm that gives input data to one of the different categories. Classification is a supervised learning process in machine learning and statistics in which a computer software takes from data input and then

performs that learns to classify input observations.

A.Neural Networks:

A Neural Network is a Network of hidden layers, input layers, and output layers that attempts to replicate the functioning of the human brain. The input data itself can be represented as an abstract representation of the hidden layers. With the help of its own internal logic, these layers enable the neural network interpret various properties of the data. These are non-interpretable neural networks. Even if the hidden layers are observed, noninterpretable models cannot be interpreted or comprehended. This is due to the fact that neural networks have their own inherent logic that we cannot fathom.

C. SVM:

SVM is machine learning technique for classifying and forecasting data.when compared to regression,classification is the most appropriate method.The SVM algorithm mainly focuses to find a hyperplane that clearly divides data points.The hyperplane size is decided by the no.of data points.If we are taking two input characteristics, the hyperplane is seems as a line.If we are taking three input features,then

B.Genetic Algorithm

Genetic Algorithm is used to detect malwares by training selected feature to ML classifiers and to check the identification capacity before and after feature selection. This experiment revealed that the genetic algorithm produces the most optimised feature output and assists in cutting the feature dimension in half. Genetic Algorithm Phases

- **Population:** A population is a group of people. The population includes the number of people being tested, information about the search space, and phenotypic parameters. In most cases, the population is started at random.
- **Individuals:** In a population, individuals are a single solution. Genes are a set of parameters that define an individual. Chromosomes are made up of genes.
- **Genes:** Genes are the fundamental components of Genetic Algorithms. Genes make up each chromosome. The solution to

the problem may be determined by the genes. They are represented by a random length bit (0 or 1) string.

• **Fitness:** The fitness of a problem's phenotype indicates its worth. The fitness function indicates how near the answer is to

being the best. The total of all parameters linked to the problem – Euclidean distance, for example – is used to determine the fitness function. There is no formula for determining fitness function.

D. Feature Selection with Genetic Algorithm

The genetic algorithm is a natural selection-based optimization tool. In this post, I demonstrate how to choose features using genetic algorithms. While scikit-learn has a number of well-known feature selection algorithms, feature selection goes far beyond what is provided there. A fundamental part of any machine learning pipeline is feature selection. However, there is an abundance of data available these days. As a result, there is frequently an excess of features. Numerous features are redundant, as is often the case with many features. They add noise to your model, making it difficult to comprehend. The issue is determining which characteristics are relevant to the issue. The goal is to have high-end features. This package is compatible with existing sklearn models and offers a lot of genetic selection features and options. For feature selection in this post, I'm utilizing a genetic algorithm. A

genetic algorithm, on the other hand, can be used to optimize hyperparameters. Because the stages are simple and broad, they can be used to a variety of situations.

E.System Architecture

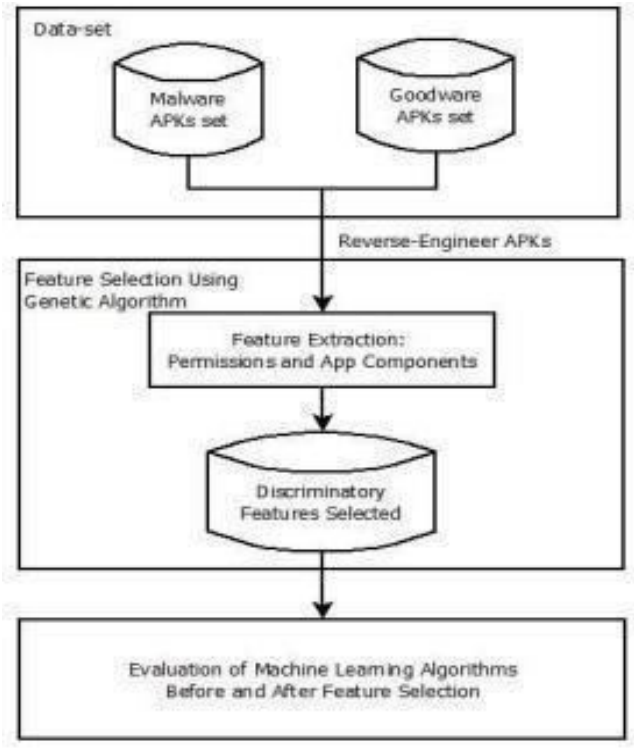
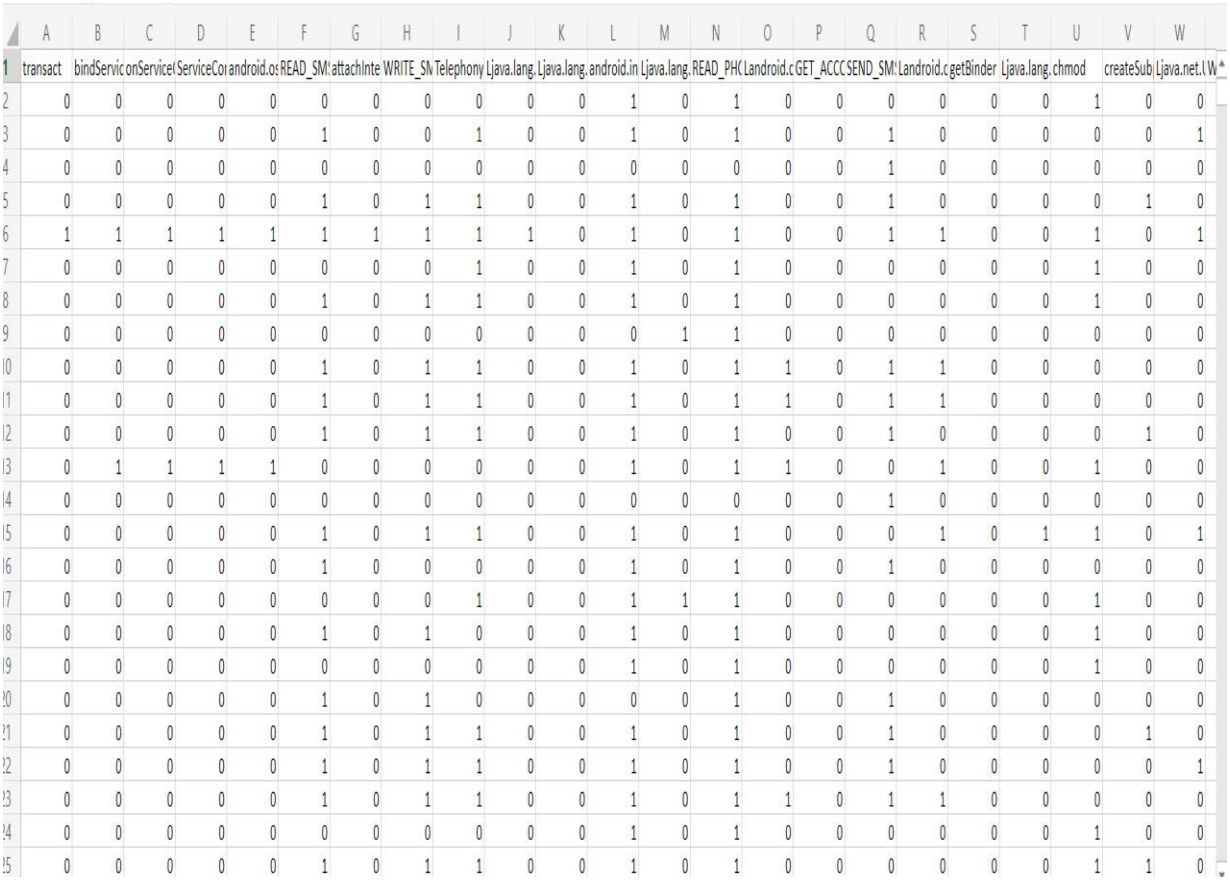


Figure 1: System Architecture

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

F.Dataset Design:

Android Dataset



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
1	transact	bindServic	onService(	ServiceCo	android.os	READ_SM	attachInte	WRITE_SN	Telephony	Ljava.lang.	Ljava.lang.	android.in	Ljava.lang.	READ_PH(	Landroid.c	GET_ACCC	SEND_SM	Landroid.c	getBinder	Ljava.lang.	chmod	createSub	Ljava.net.U	W
2	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	
3	0	0	0	0	0	1	0	0	1	0	0	1	0	1	0	0	1	0	0	0	0	0	1	
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	
5	0	0	0	0	0	1	0	1	1	0	0	1	0	1	0	0	1	0	0	0	0	1	0	
6	1	1	1	1	1	1	1	1	1	1	0	1	0	1	0	0	1	1	0	0	1	0	1	
7	0	0	0	0	0	0	0	0	1	0	0	1	0	1	0	0	0	0	0	0	1	0	0	
8	0	0	0	0	0	1	0	1	1	0	0	1	0	1	0	0	0	0	0	0	1	0	0	
9	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	
10	0	0	0	0	0	1	0	1	1	0	0	1	0	1	1	0	1	1	0	0	0	0	0	
11	0	0	0	0	0	1	0	1	1	0	0	1	0	1	1	0	1	1	0	0	0	0	0	
12	0	0	0	0	0	1	0	1	1	0	0	1	0	1	0	0	1	0	0	0	0	1	0	
13	0	1	1	1	1	0	0	0	0	0	0	1	0	1	1	0	0	1	0	0	1	0	0	
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	
15	0	0	0	0	0	1	0	1	1	0	0	1	0	1	0	0	0	1	0	1	1	0	1	
16	0	0	0	0	0	1	0	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	
17	0	0	0	0	0	0	0	0	1	0	0	1	1	1	0	0	0	0	0	0	1	0	0	
18	0	0	0	0	0	1	0	1	0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	
19	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	
20	0	0	0	0	0	1	0	1	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	
21	0	0	0	0	0	1	0	1	1	0	0	1	0	1	0	0	1	0	0	0	0	1	0	
22	0	0	0	0	0	1	0	1	1	0	0	1	0	1	0	0	1	0	0	0	0	0	1	
23	0	0	0	0	0	1	0	1	1	0	0	1	0	1	1	0	1	1	0	0	0	0	0	
24	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	
25	0	0	0	0	0	1	0	1	1	0	0	1	0	1	0	0	0	0	0	0	1	1	0	

Figure 2: Android Dataset

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

6. Experimental Result:

SVM:

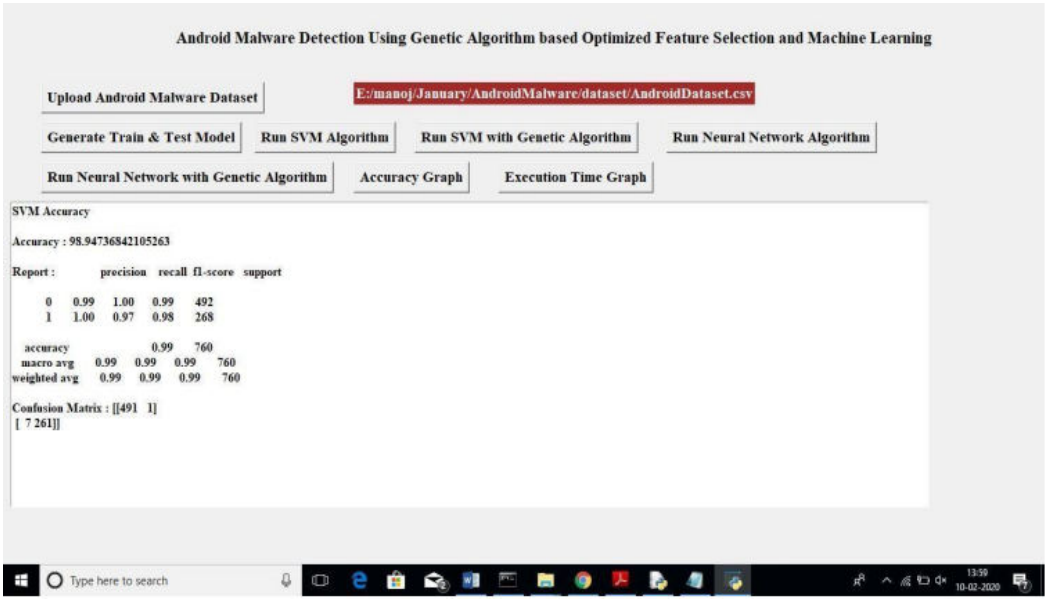


Figure 3: SVM Accuracy of Android Data Set

SVM With GA:

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

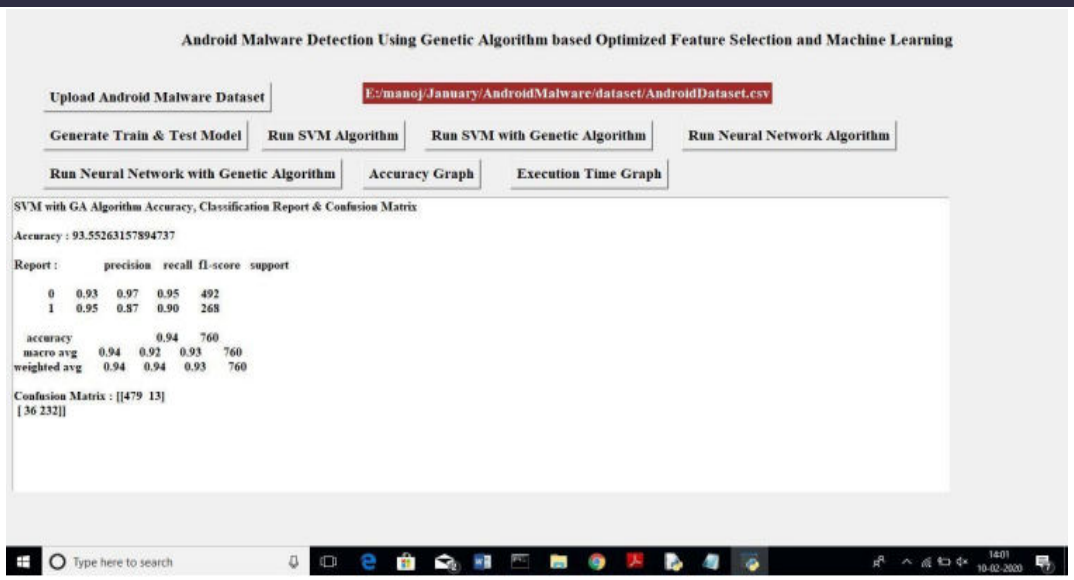


Figure 4 :SVM With GA Accuracy of Android Data Set

ANN:

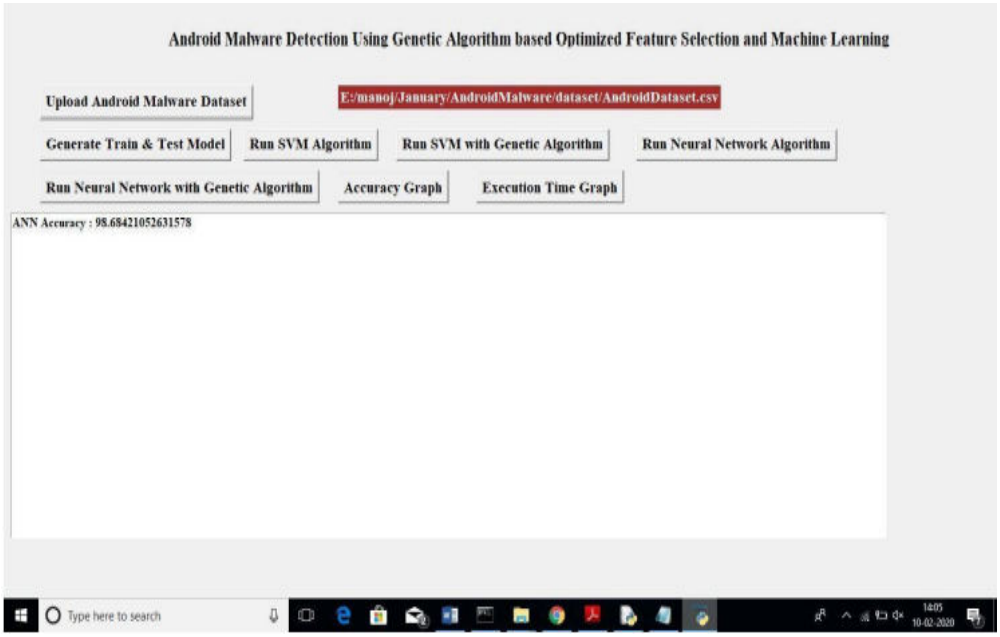


Figure 5:ANN Accuracy of Android Data Set

ANN With GA:

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

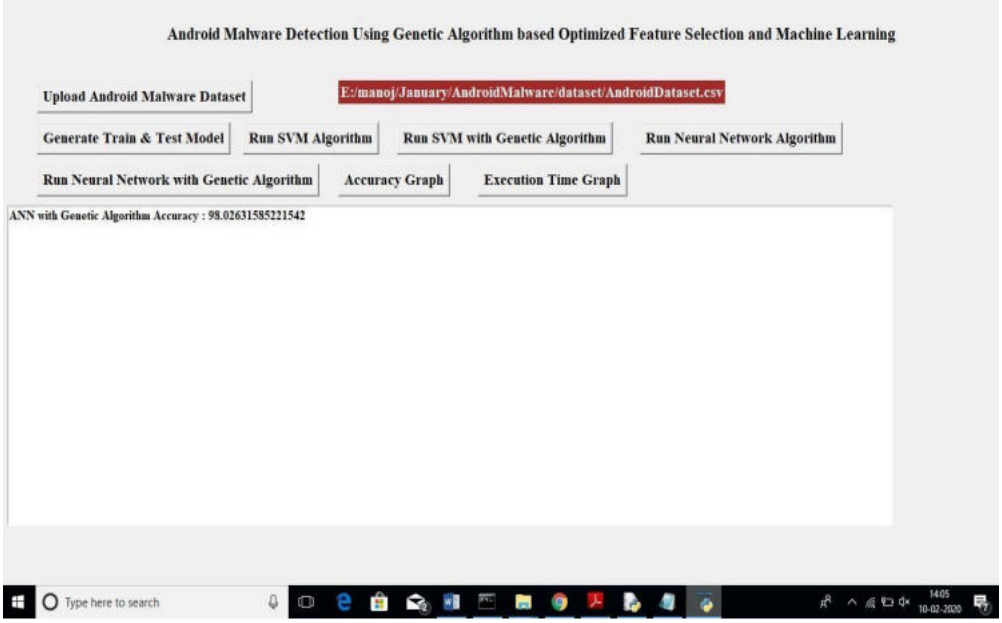
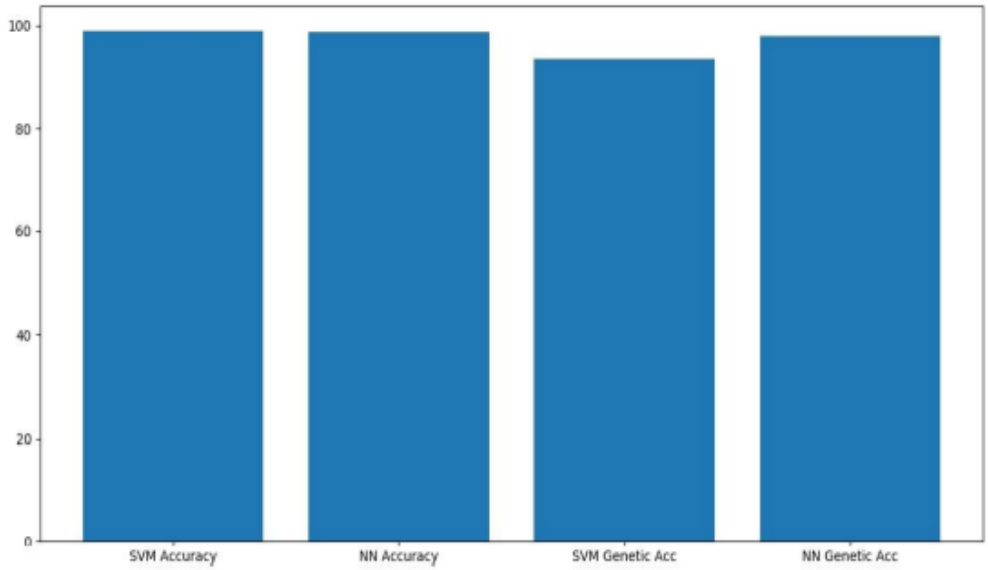


Figure 6: ANN With GA Accuracy of Android Data Set

Accuracy Graph:

The x-axis shows the algorithm name, while the y-axis reflects the accuracy, and SVM has a high accuracy overall. To receive the execution time of all algorithms, click the 'Execution Time



n.

FIGURE 7: Comparison of svm, nn, svm genetic, nn genetic accuracy on Android dataset.

Execution Time Graph:
The x-axis in the graph above reflects the algorithm name, while the y-axis represents the execution time. We can deduce from the graph above that machine learning methods using genetic algorithms.

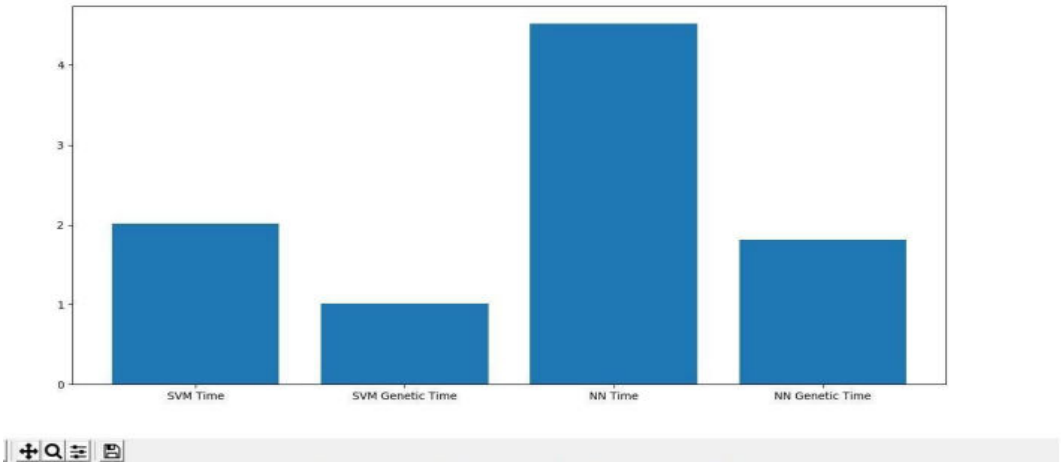


FIGURE 8: Comparison of svm, svm genetic, nn, nn genetic execution times on android dataset.

7.Performance Evaluation

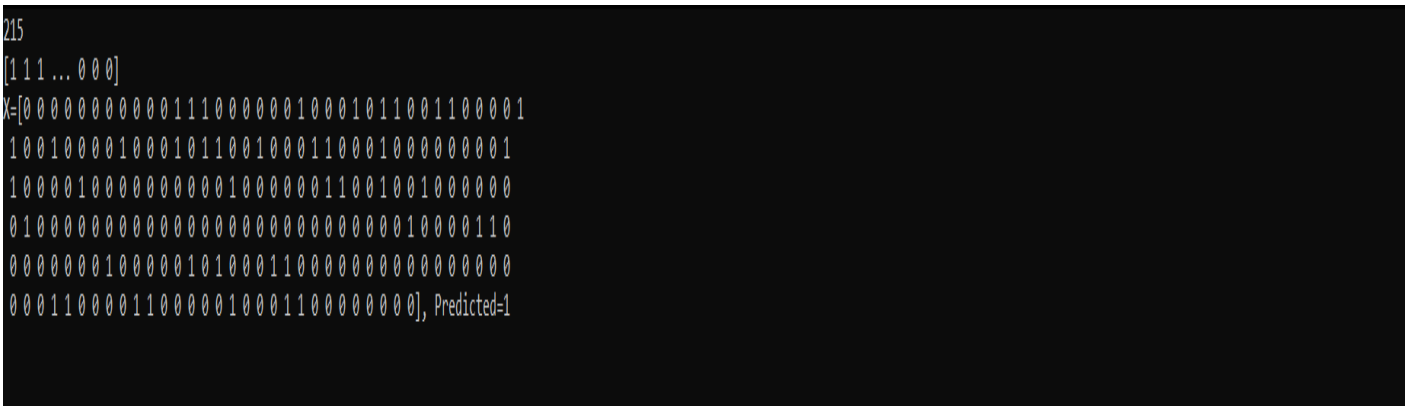


FIGURE 9:SVM matrix prediction on android dataset.

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

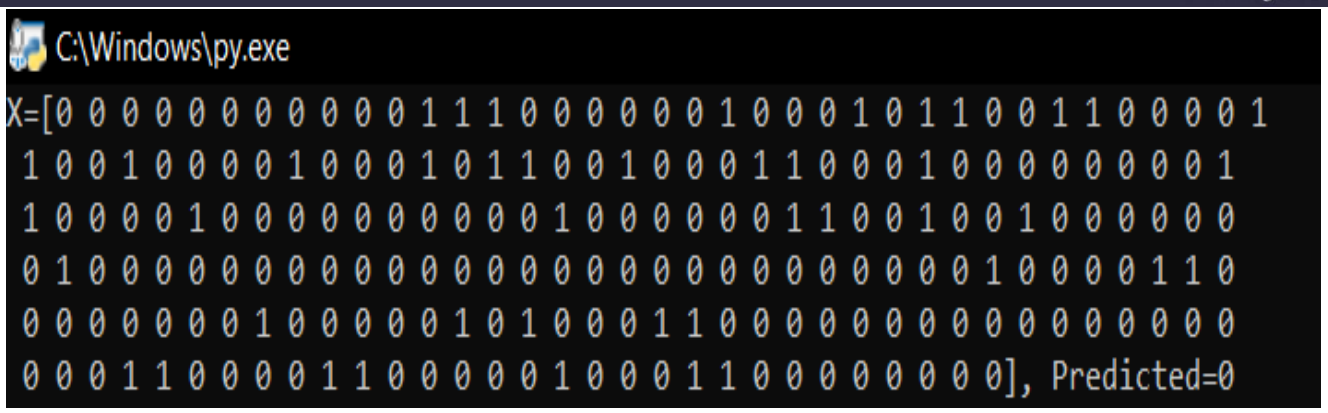


FIGURE 10:SVM with GA matrix prediction on android dataset.

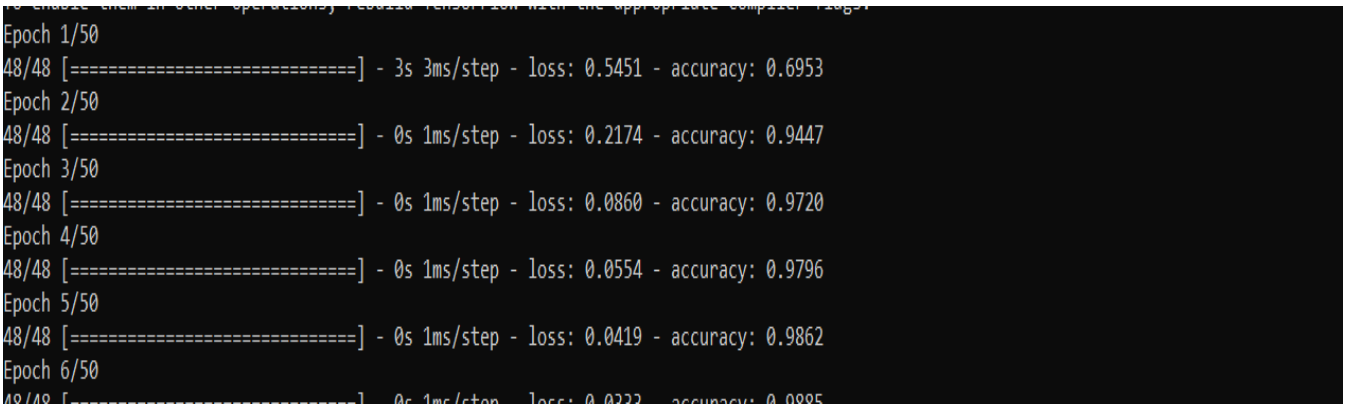


FIGURE 11: NN initial epoch on android dataset.

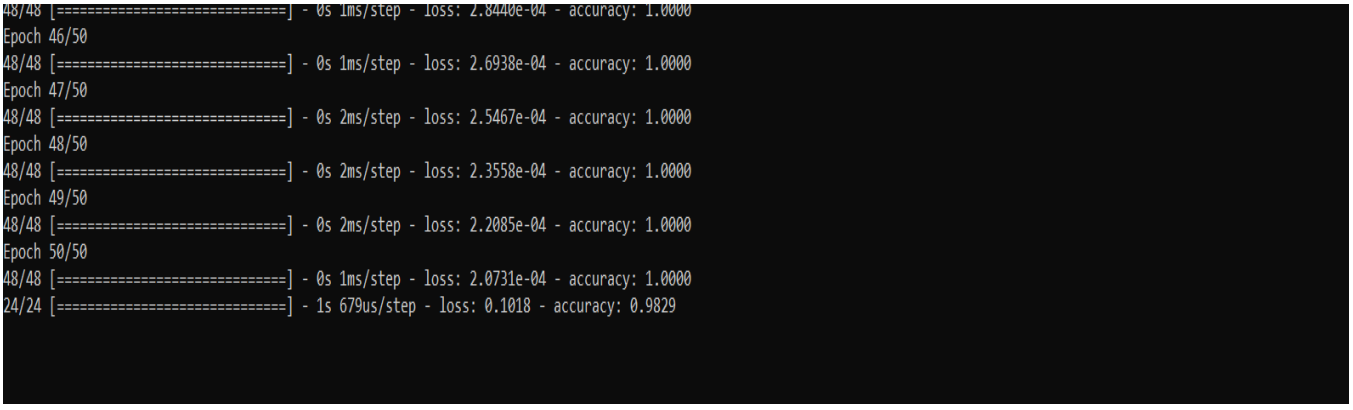


FIGURE 12: NN final epoch on android dataset.

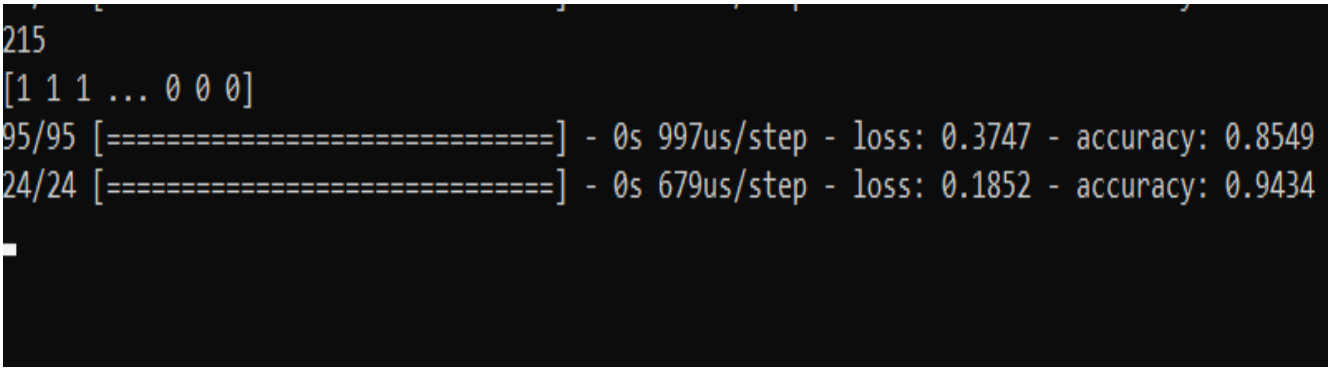


FIGURE 13: NN with GA on android dataset.

8.Conclusion

As the amount of threats offered to Android systems expands every day, it's critical to create a framework that can effectively detect malware. The majority of these risks are spread by malicious software or malware. Machine learning-based approaches are utilised when signature-based approaches fail to detect new versions of malware posing zero-day dangers. To find the best efficient feature subset that may be utilised to effectively train machine learning algorithms, the suggested methodology uses an evolving Genetic Algorithm. Experiments show that utilising Support Vector Machine and Neural Network classifiers on smaller dimension feature sets can retain a reasonable classification accuracy of more than 94 percent while reducing the classifiers' training complexity. Using larger datasets and assessing the impact of different machine learning algorithms

9. References

[1]Unver, H.M., Bakour, K. Android malware detection based on image-based features and machine learning techniques. SN Appl. Sci. 2, 1299 (2020). https://doi.org/10.1007/s42452-020-3132-2

[2] H. Han, S. Lim, K. Suh, S. Park, S. -j. Cho and M. Park, "Enhanced Android Malware Detection: An SVM-Based Machine Learning Approach," 2020 IEEE International Conference on Big Data and Smart Computing (BigComp), 2020, pp. 75-81, doi: 10.1109/BigComp48618.2020.00-96.

[3] D. Wu, C. Mao, T. Wei, H. Lee and K. Wu, "DroidMat: Android Malware Detection through Manifest and API Calls Tracing," 2012 Seventh Asia Joint Conference on Information Security, 2012, pp. 62-69, doi: 10.1109/AsiaJCIS.2012.18.

[4] Hyo-Sik Ham and Mi-Jung Choi, "Analysis of Android malware detection performance using machine learning classifiers," 2013 International Conference on ICT Convergence (ICTC), 2013, pp. 490-495, doi: 10.1109/ICTC.2013.6675404.

[5]J. Sahs and L. Khan, "A Machine Learning Approach to Android Malware Detection," 2012 European Intelligence and Security

Informatics Conference, 2012, pp. 141-147, doi: 10.1109/EISIC.2012

[6]D. Arp, M. Spreitzenbarth, M. Hübner, H. Gascon, and K. Rieck, "Drebin: Effective and Explainable Detection of Android Malware in Your Pocket," in Proceedings 2014 Network and Distributed System Security Symposium, 2014.

[7] N. Milosevic, A. Dehghantanha, and K. K. R. Choo, "Machine learning aided Android malware classification," Comput. Electr. Eng., vol. 61, pp. 266–274, 2017.

[8] J. Li, L. Sun, Q. Yan, Z. Li, W. Srisa-An, and H. Ye, "Significant Permission Identification for MachineLearning-Based Android Malware Detection," IEEE Trans. Ind. Informatics, vol. 14, no. 7, pp. 3216–3225, 2018.

[9]A. Saracino, D. Sgandurra, G. Dini, and F. Martinelli, "MADAM: Effective and Efficient Behaviorbased Android Malware Detection and Prevention," IEEE Trans. Dependable Secur. Comput., vol. 15, no. 1, pp. 83–97, 2018.