



ANDROID MALWARE DETECTION USING GENETIC ALGORITHM BASED ON OPTIMIZED FEATURE SELECTION AND MACHINE LEARNING

K. Preethi Sai, V. Navya, T. Sreya, Sk. Sajid Ali

UG students, Dept of CSE, Kallam Haranadhareddy Institute of Technology, AP, India.

ABSTRACT

Android platform due to opensource characteristic and Google backing has the largest global market share. Being the world's most popular operating system, it has drawn the attention of cyber criminals operating particularly through wide distribution of malicious applications. This paper proposes an effectual machine-learning based approach for Android Malware Detection making use of evolutionary Genetic algorithm for discriminatory feature selection. Selected features from Genetic algorithm are used to train machine learning classifiers and their capability in identification of Malware before and after feature selection is compared. The experimentation results validate that Genetic algorithm gives most optimized feature subset helping in reduction of feature dimension to less than half of the original feature-set. Classification accuracy of more than 94% is maintained post feature selection for the machine learning based classifiers, while working on much reduced feature dimension, thereby, having a positive impact on computational complexity of learning classifiers.

Keywords: Android, Cyber criminals, Malicious, Genetic Algorithm, Malware, Detection, Machine learning.

1. INTRODUCTION

Android Apps are freely accessible on Google Play store, the official Android app store as well as third party app shops for consumers to download. Due to its opensource nature and popularity, malware authors are increasingly concentrating on creating harmful apps for Android operating system. In spite of various attempts by Google Play store to protect against malicious apps, they still find their way to mass market and cause harm to users by misusing personal information related to their phone book, mail accounts, GPS location information and others for misuse by third parties or else take control of the phones remotely. Android Apps are freely accessible on Google Play store, the official Android app store as well as third-party app shops for consumers to download. Due to its opensource nature and popularity, malware authors are increasingly concentrating on creating harmful apps for Android operating system. In spite of various attempts by Google Play store to protect against malicious apps, they still find their way to mass market and cause harm to users by misusing personal information related to their phone book, mail accounts, GPS location information and others for misuse by third parties or else take control of the phones remotely. Therefore, there is need to conduct malware analysis or reverse-engineering of such harmful apps which represent significant danger to Android systems. Broadly speaking, Android Malware analysis is of two types: Static Analysis and Dynamic Analysis. Static analysis essentially includes evaluating the code structure without running it whereas dynamic analysis is evaluation of the runtime behaviour of Android Apps under restricted environment. Given in to the ever-increasing varieties of Android Malware presenting zero-day risks, an effective method for detection of Android malwares is needed. In contrast to signature-based method which needs frequent updating of signature database, machine learning based approach in conjunction with static and dynamic analysis may be used to identify new variants of Android Malware presenting zero-day risks. Android is an open-source free operating system and it has support from Google to publish android application on its Play Store. Anybody can develop an



android app and publish on play store free of cost. This android feature attracts cyber-criminals to developed and publish malware app on play store. If anybody install such malware app then it will steal information from phone and transfer to cyber-criminals or can give total phone control to criminal's hand. To protect users from such app in this paper author is using machine learning algorithm to detect malware from mobile app.

2. LITERATURE SURVEY

Mobile gadgets, such as smartphones, iPads, and computer tablets, have become daily needs to accomplish essential activities, including education, paying bills online, bank transactions, employment information, and pleasure. Based on the information from an online mobile device manufacturing website, Android is one of the prominent operating systems (OS) utilized by manufacturers (Rayner, 2019; J kietly, 2019). The opensource framework of Android has helped the smartphone makers in creating Android devices of different sizes and kinds, such as smartphones, smart watches, smart TVs, and smart eyewear. In the most recent decades, the amount of amazing Android devices accessible globally has grown from 38 in 2009 to over 20,000 in 2016 (Android, 2019). As a consequence of the demand for this Android OS, the latest data from Statista showed that the number of Android malware rise to 26.6 million in March 2018 .(Statista, 2019). Moreover, McAfee identified a virus known as Garbo's, which exploits the Android and breaks Google Play Store security (McAfee, 2019). (McAfee, 2019). It was also estimated that 17.5 million Android devices have downloaded this Garbo's mobile virus before they were taken down. Mobile malware is intended to disable a mobile device, enable malevolent actions to remotely control the device, or steal personal information (Beal, 2013). (Beal, 2013). Moreover, these harmful actions able to execute silently and circumvent permission if the Android kernel is infected by mobile malware (Ma & Sharaf, 2013; Aubrey-Derrick Schmidt et al., 2009). In September 2019, a total of 172 harmful apps were discovered on Google Play Store, with roughly 330 million installs. According to experts, the harmful components were concealed within the functioning apps. After the apps are downloaded, it leads to the display of popup advertising, which continue visible even when the application was closed (O'Donnell, 2019). To identify this virus, security practitioners performing malware analysis, which attempts to analyse the malicious features and behaviours. There are dynamic, static, and hybrid analyses. To merge the features of the static and dynamic approach, three-layer detection model called SAM Android has been developed by Saba Arshad et al. (2018) which integrates static and dynamic properties. Mobile Sandbox by spreitzenbarthet al. (2015) which suggested to utilize the findings of static analysis to drive the dynamic analysis and ultimately achieve categorization. The hybrid analysis method is excellent to assist in increasing the accuracy, but it also has a significant disadvantage such as the waste of time and space for the large number of malware samples to be identified and analysed (Fang et al., 2020; Alswaina & Elleithy, 2020). Notably, our review post included additional elements than authorization by adding an in-depth emphasis on static analysis. Meanwhile, the second study, analyses three anti-viruses, namely Stowaway, AASandbox, and Droid box. To help in the choice about the best anti-virus, the aforementioned anti-viruses were separated using static and dynamic analyses, followed by a comparison between one another (Ma & Sharbat, 2013).

3. PROPOSED SYSTEM

Two set of Android Apps or APK's: Malware/Good ware reverse engineered to extract features such as permissions and count of App component such as Activity, Services, Content Providers. These features are used as feature vector with class labels as Malware and Good ware represented by 0 and 1

respectively in CSV format. To reduce dimensionality of feature-set, the CSV is fed to Genetic Algorithm to select the most optimized set of features. The optimized set of features obtained is used for training two machine learning classifiers: Support vector machine and Neural Network. In the proposed methodology, static features are obtained from AndroidManifest.xml which contains all the important information needed by any Android platform about Apps. Andro guard tool has been used for disassembling of the APK's and getting the static features.

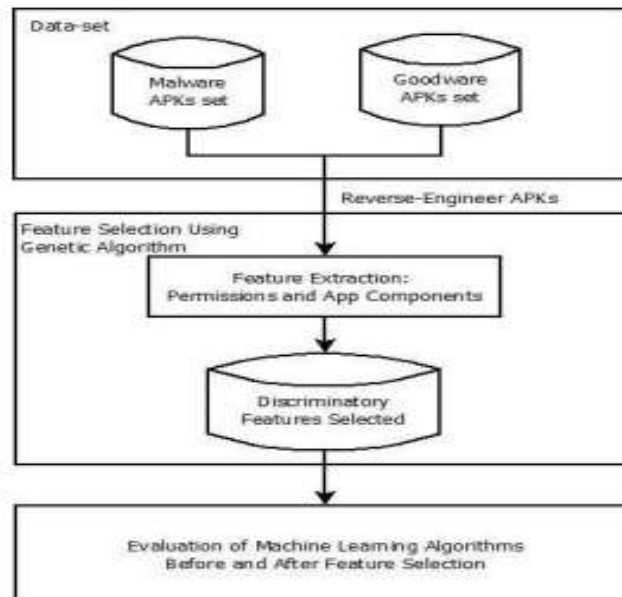


Fig. 1. Proposed Methodology

4. Existing System

The major contribution of the study is reduction of feature dimension to less than half of original feature-set using Genetic Algorithm such that it can be given as input to machine learning classifiers for training with decreased complexity while retaining their accuracy in malware classification. In contrast to exhaustive technique of feature selection which involves testing for 2^N distinct combinations, where N is the number of features, Genetic Algorithm, a heuristic searching strategy based on fitness function has been employed for feature selection. The optimal feature set generated via Genetic algorithm is utilized to train two machine learning algorithms: Support Vector Machine and Neural Network. It is found that a reasonable classification accuracy of more than 94 percent is maintained despite working on a considerably smaller feature dimension, thus, decreasing the training time complexity of classifiers.

5. Algorithms used in this project

5.1 Genetic Algorithm:

A genetic algorithm is an adaptive heuristic search algorithm inspired by "Darwin's theory of evolution in Nature." It is used to solve optimization problems in machine learning. It is one of the important algorithms as it helps solve complex problems that would take a long time to solve. Genetic Algorithms are being widely used in different real-world applications, for example, Designing electronic circuits, code-breaking, image processing, and artificial creativity. Before understanding the Genetic algorithm, let's first understand basic terminologies to better understand this algorithm.

The steps involved in feature selection using Genetic Algorithm can be summarized as below:

Step 1: Initialize the algorithm using feature subsets which are binary encoded such that if the feature is included it is represented by 1 and if it is excluded it is represented by 0 in the chromosome.

Step 2: Start the algorithm defining an initial set of population generated randomly.

Step 3: Assign a fitness score calculated by the defined fitness function for genetic algorithm.

Step 4: Selection of Parents: Chromosomes with good fitness scores are given preference over others to produce next generation of off-springs.

Step 5: Perform crossover and mutation operations on the selected parents with the given probability of crossover and mutation for generation of off-springs.

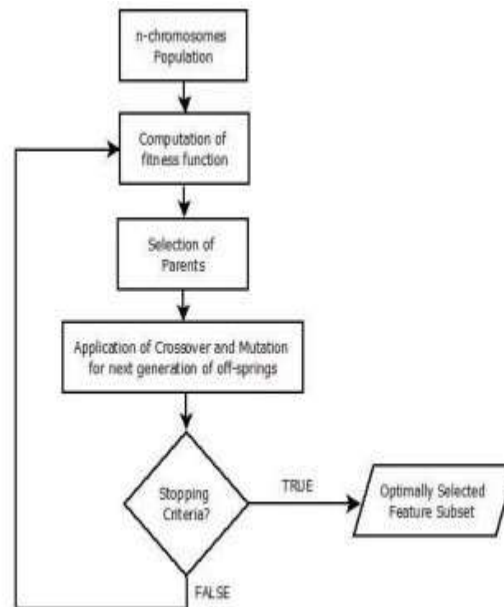


Fig 2: Genetic Algorithm

5.2 Machine Learning Algorithm

Machine learning refers to computer programming used to optimize the performance criteria using example data or experience. T. M. Mitchell described machine learning as inherently multidisciplinary field used in various real-world applications. Machine learning is conducted by introducing several algorithms to solve problems in various fields. This is the reason why machine learning must be evaluated through various algorithms. Thus, we selected nine algorithms to evaluate the performance of feature selection.

5.2.1. Support Vector Machine

Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning. The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane. SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine.

5.2.2. Neural Network

The goal of machine learning is to develop computer programs that can use data to learn by themselves. Machine learning achieves this by utilizing neural networks modeled loosely after the structure of the biological brain. The human brain consists of a network of neurons responsible for



creating new connections in the brain, thus creating new memories and recording learned information. These neural networks are also responsible for retrieving information and using it to recognize patterns. Neural networks in machine learning refer to a set of algorithms designed to help machines recognize patterns without being explicitly programmed. They consist of a group of interconnected nodes. These nodes represent the neurons of the biological brain.

5.3. Feature selection technique in Machine Learning

While developing the machine learning model, only a few variables in the dataset are useful for building the model, and the rest features are either redundant or irrelevant. If we input the dataset with all these redundant and irrelevant features, it may negatively impact and reduce the overall performance and accuracy of the model. Hence it is very important to identify and select the most appropriate features from the data and remove the irrelevant or less important features, which is done with the help of feature selection in machine learning. Before implementing any technique, it is important to understand, need for the technique and so for the Feature Selection.

As we know, in machine learning, it is necessary to provide a pre-processed and good input dataset in order to get better outcomes. We collect a huge amount of data to train our model and help it to learn better. Generally, the dataset consists of noisy data, irrelevant data, and some part of useful data. Moreover, the huge amount of data also slows down the training process of the model, and with noise and irrelevant data, the model may not predict and perform well. So, it is very necessary to remove such noises and less-important data from the dataset and to do this, and Feature selection techniques are used. Selecting the best features helps the model to perform well. For example, suppose we want to create a model that automatically decides which car should be crushed for a spare part, and to do this, we have a dataset. This dataset contains a Model of the car, Year, Owner's name, Miles. So, in this dataset, the name of the owner does not contribute to the model performance as it does not decide if the car should be crushed or not, so we can remove this column and select the rest of the features(column) for the model building.

5.4 Experimental Methodology

This As the number of malware threats is increasing, it becomes increasingly important to protect our computers and smartphones with anti-malware software. Machine learning is a powerful technology for detecting malicious software. It is trained on millions of samples so that it can learn to spot their characteristics at scale even when there are new types of malwares which have never been seen before. It is an approach to artificial intelligence that can be used to detect malware which uses pattern recognition, which extracts features from the file and compares them to known malware signatures. Also, it includes scanning the entire system or parts of it, extracting features of malicious software, comparing these features to known behaviours, and identifying the presence of malware. There is a very large competition between malware designer and examiner. Both research communities are working similarly, one of them designer is developing the malware detection system and other is designing the malicious software detect the software which will attack the computer and networks resources. Malware examiner examines the known malware and try to detect the malware to avoid the attack on the user's computer systems. Malware found in the user's system are detected using either signature-based or behaviour-based techniques. The signature-based malware detection system is quick and systematic but can be easily avoided by the obfuscated malware. On the other side, behaviour-based techniques are stronger as compared to the obfuscation technique. Although, behaviour-based techniques are very time-consuming. The onus for protection on the data is on the user. There are many different types of malwares which can attack in different ways. For example, spyware can record keystrokes and further make hacker's way much easier as the spyware aims to



gather data about the victim or any organization and forward it to another system in such a way that it breaches the victim's privacy and benefits the cyber punks. While ransomware is a malware that encrypts user's computer files and demands ransom for data to be unlocked again. A user or organization's censorious data is encoded so that they cannot access databases, files, or applications. Hence, some amount of money is demanded to provide access for their data only. System virus is generally a program that is built outside the user's will and can cause damage to both the operating system and the hardware (physical) elements of a system. The various effects caused due the viruses are:

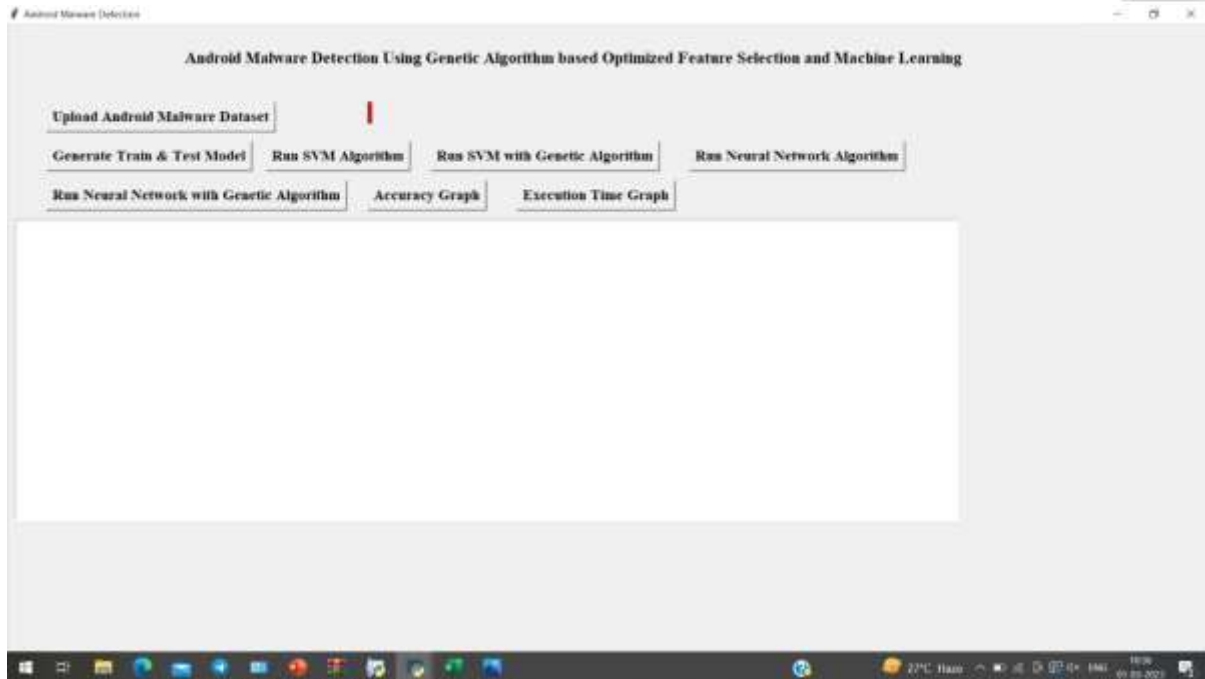
- [1] Destruction of files in the system.
 - [2] Change in the file size.
 - [3] Delete all data on the disc.
 - [4] Damage in the file allocation table, which makes it impossible to read the information on the disk.
 - [5] Various innocuous but disturbing graphic / sound effects.
 - [6] Slowing down the working speed of the computer until it crashes worms.
- Computer worms are programs with damaging effects which use communication between computers to spread. Worms have similar features with viruses, worms are able spread like viruses in the system, but not locally, but on other computers. It uses computer networks to spread to other systems.

6. RESULTS AND DISCUSSION

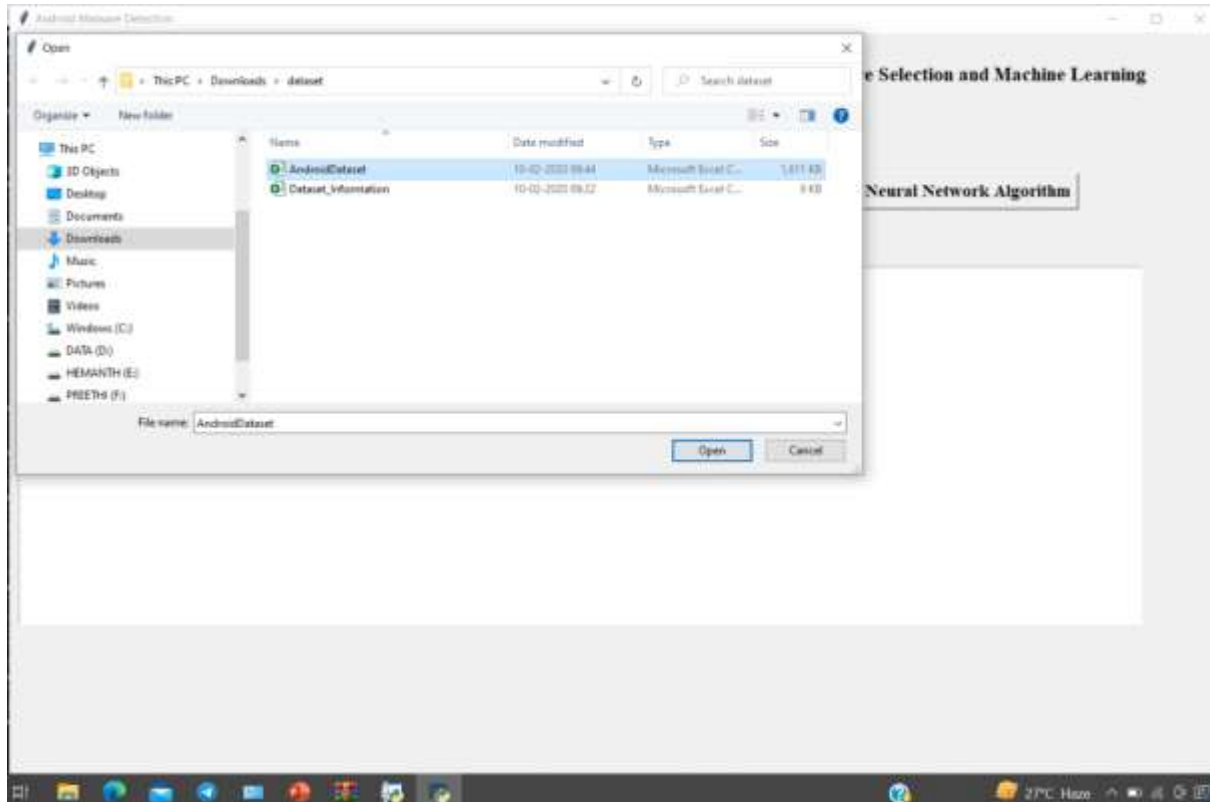
To train the existing and proposed models, this project has used 'Android Malware Dataset'.

To implement this project, we have designed following modules.

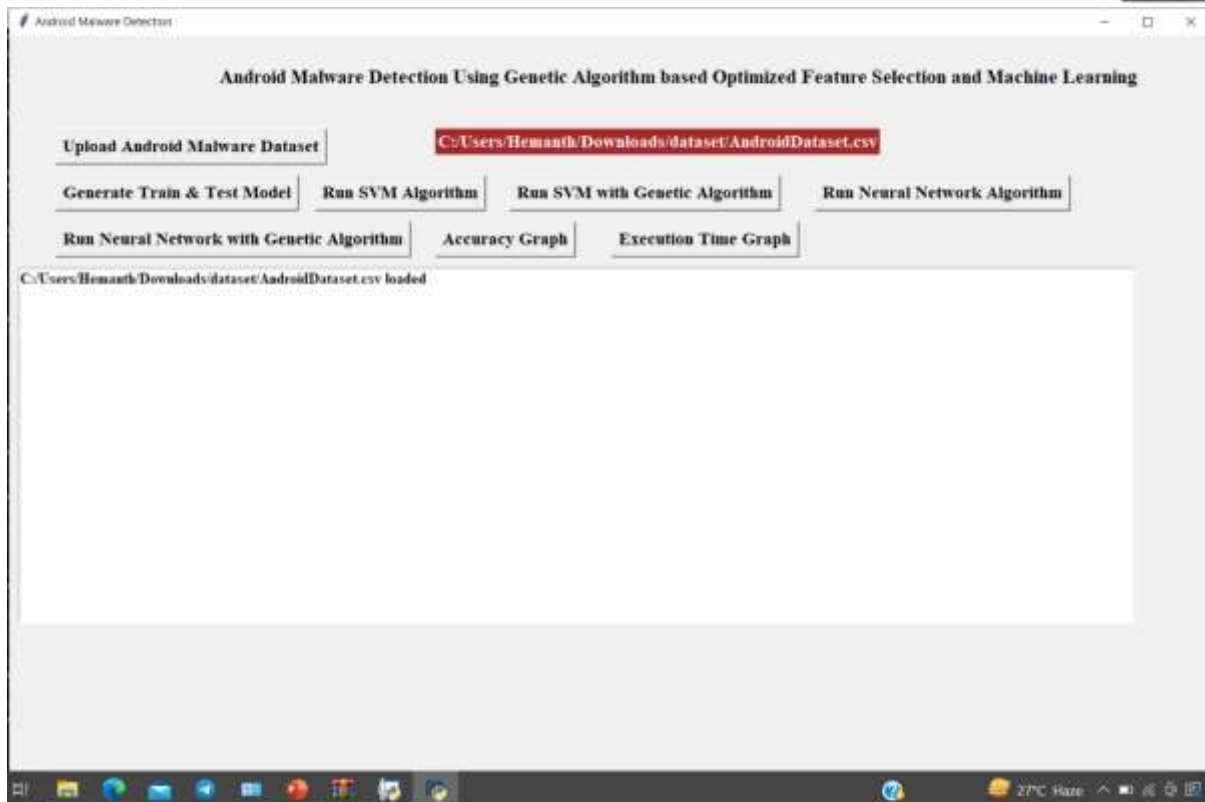
- 1) Upload Android Malware Dataset: using this module we will upload Androiddataset.
- 2) Generate Train & Test Model : using this button to split dataset into train and test part.
- 3) Run SVM Algorithm: using this button to generate SVM model on train and test and get its accuracy
- 4) Run SVM with Genetic Algorithm: using this button to choose optimize features and then run SVM on optimize features to get accuracy
- 5) Run Neural Network Algorithm: using this button to test neural network accuracy.
- 6) Run Neural Network with Genetic Algorithm: using this button to get NN accuracy with genetic algorithm.
- 7) Accuracy Graph: using this button to see all algorithms accuracy in graph.
- 8) Execution Time Graph: using this button to get execution time of all algorithm.



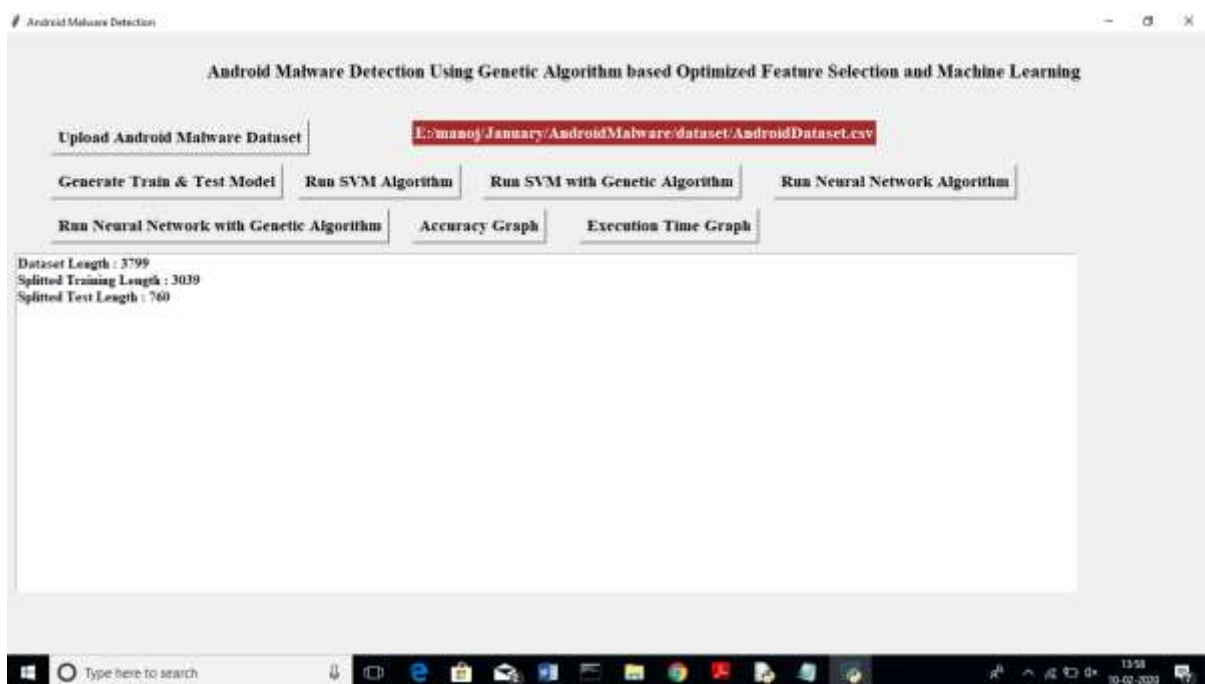
In above screen click on 'Upload Android Malware Dataset' button and upload dataset.



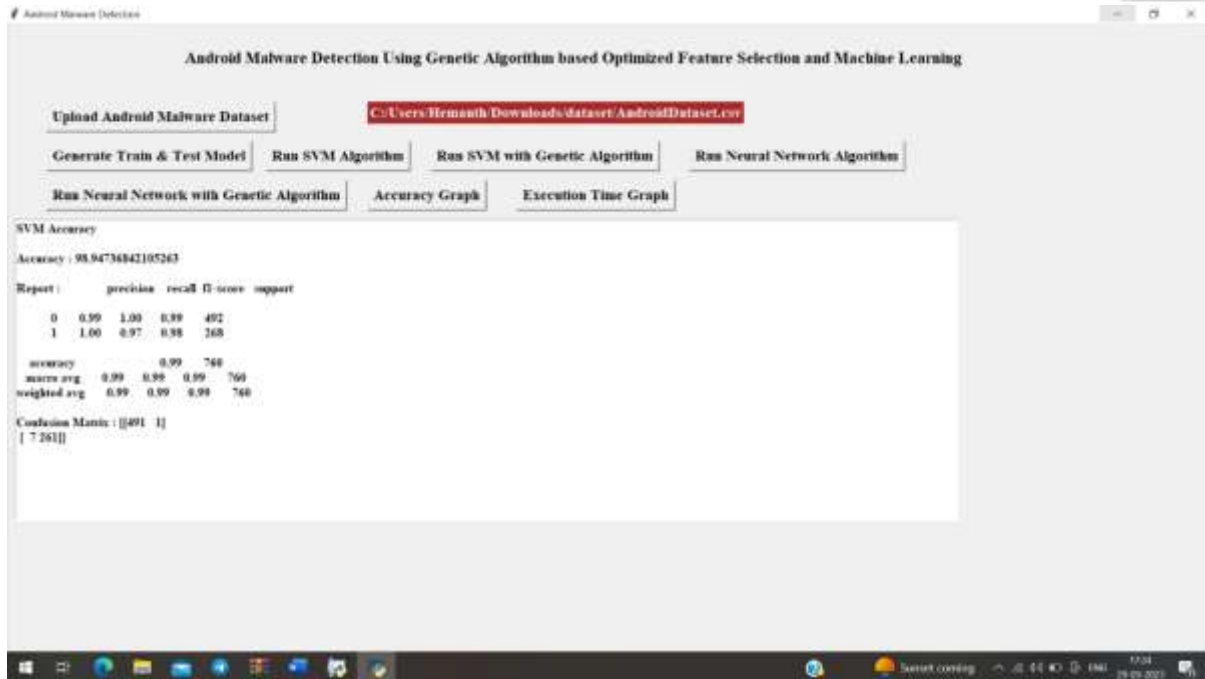
In above screen I am uploading 'AndroidDataset.csv' file and after upload will get below screen



Now click on 'Generate Train & Test Model' button to split dataset into train and test part. All machine learning algorithms will take 80% dataset for training and 20% dataset to test accuracy of trained model. After clicking that button will get train and test model.



In above screen we can see there are total 3799 android app records are there and application using 3039 records for training and 760 records for testing. Now we have both train and test model and now click on 'Run SVM Algorithm' button to generate SVM model on train and test and get its accuracy

Android Malware Detection Using Genetic Algorithm based Optimized Feature Selection and Machine Learning

Upload Android Malware Dataset: C:\Users\Hermanth\Downloads\dataset\AndroidDataset.csv

Generate Train & Test Model | Run SVM Algorithm | Run SVM with Genetic Algorithm | Run Neural Network Algorithm

Run Neural Network with Genetic Algorithm | Accuracy Graph | Execution Time Graph

SVM Accuracy

Accuracy : 98.94738842105263

Report :

	precision	recall	F1-score	support
0	0.99	1.00	0.99	492
1	1.00	0.97	0.98	268

accuracy 0.99 0.99 760

macro avg 0.99 0.99 0.99 760

weighted avg 0.99 0.99 0.99 760

Confusion Matrix : [[491 1]
[268]]

In above screen we got 98% accuracy for SVM and now click on ‘Run SVM with Genetic Algorithm’ button to choose optimize features and then run SVM on optimize features to get accuracy



Android Malware Detection Using Genetic Algorithm based Optimized Feature Selection and Machine Learning

Upload Android Malware Dataset: E:\manoj\January\AndroidMalware\dataset\AndroidDataset.csv

Generate Train & Test Model | Run SVM Algorithm | Run SVM with Genetic Algorithm | Run Neural Network Algorithm

Run Neural Network with Genetic Algorithm | Accuracy Graph | Execution Time Graph

SVM with GA Algorithm Accuracy, Classification Report & Confusion Matrix

Accuracy : 93.55263157894737

Report :

	precision	recall	F1-score	support
0	0.93	0.97	0.95	492
1	0.95	0.87	0.90	268

accuracy 0.94 760

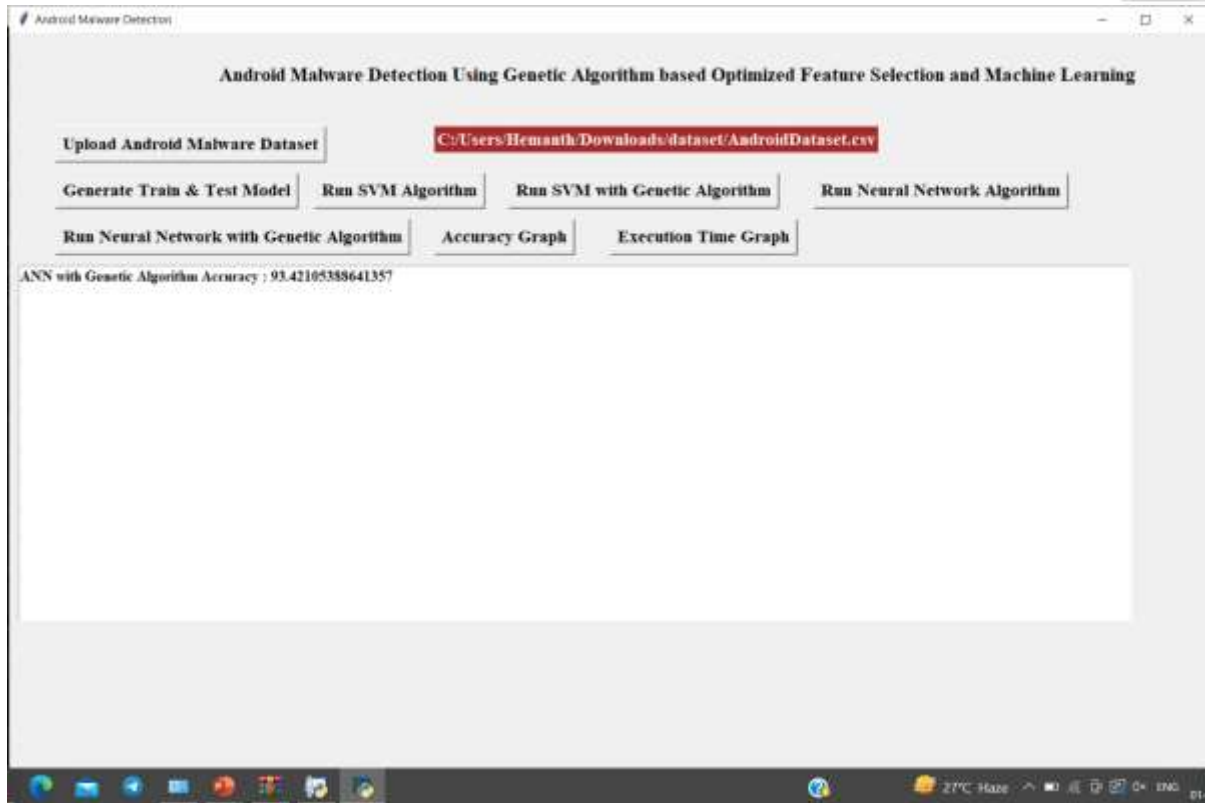
macro avg 0.94 0.92 0.93 760

weighted avg 0.94 0.94 0.93 760

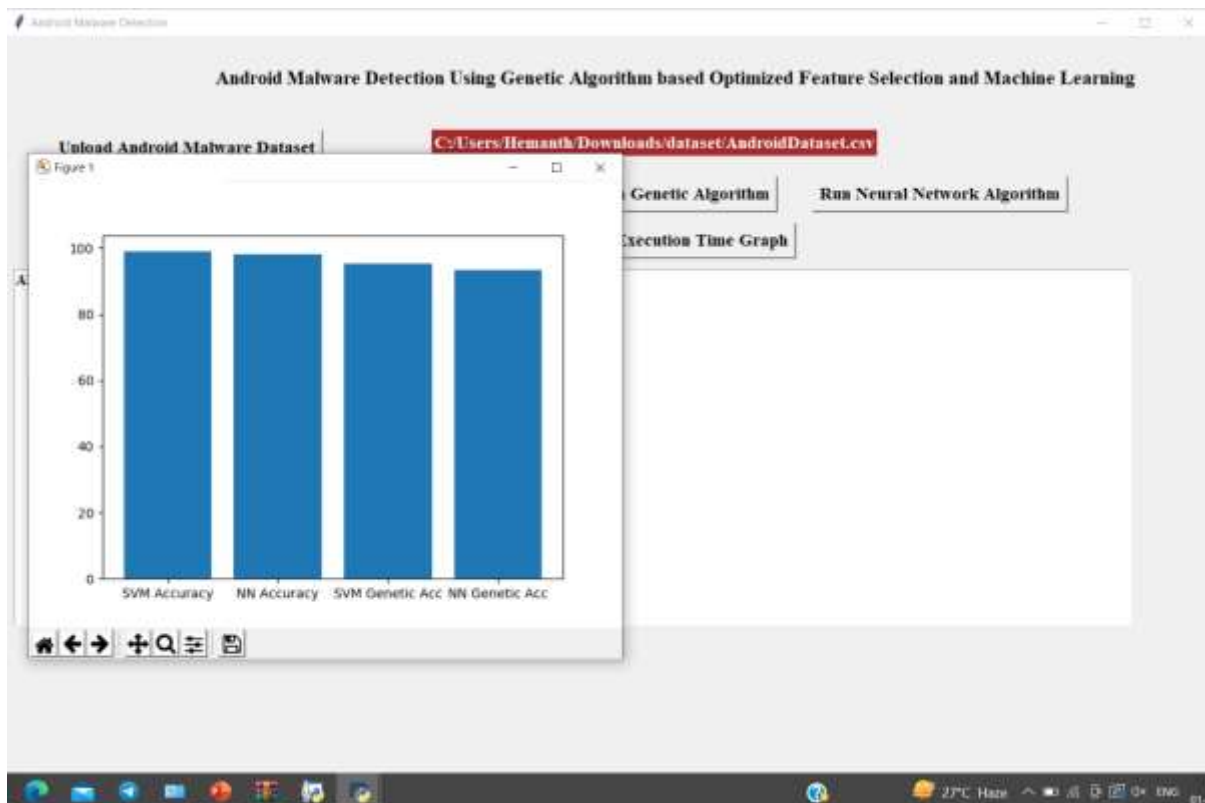
Confusion Matrix : [[479 13]
[36 232]]

In above screen SVM with Genetic algorithm got 93% accuracy. Genetic with SVM accuracy is less but its execution time will be less which we can see at the time of comparison graph.

(Note: when u run genetic then 4 empty windows will open u just close all those 4 windows and let main window to run)

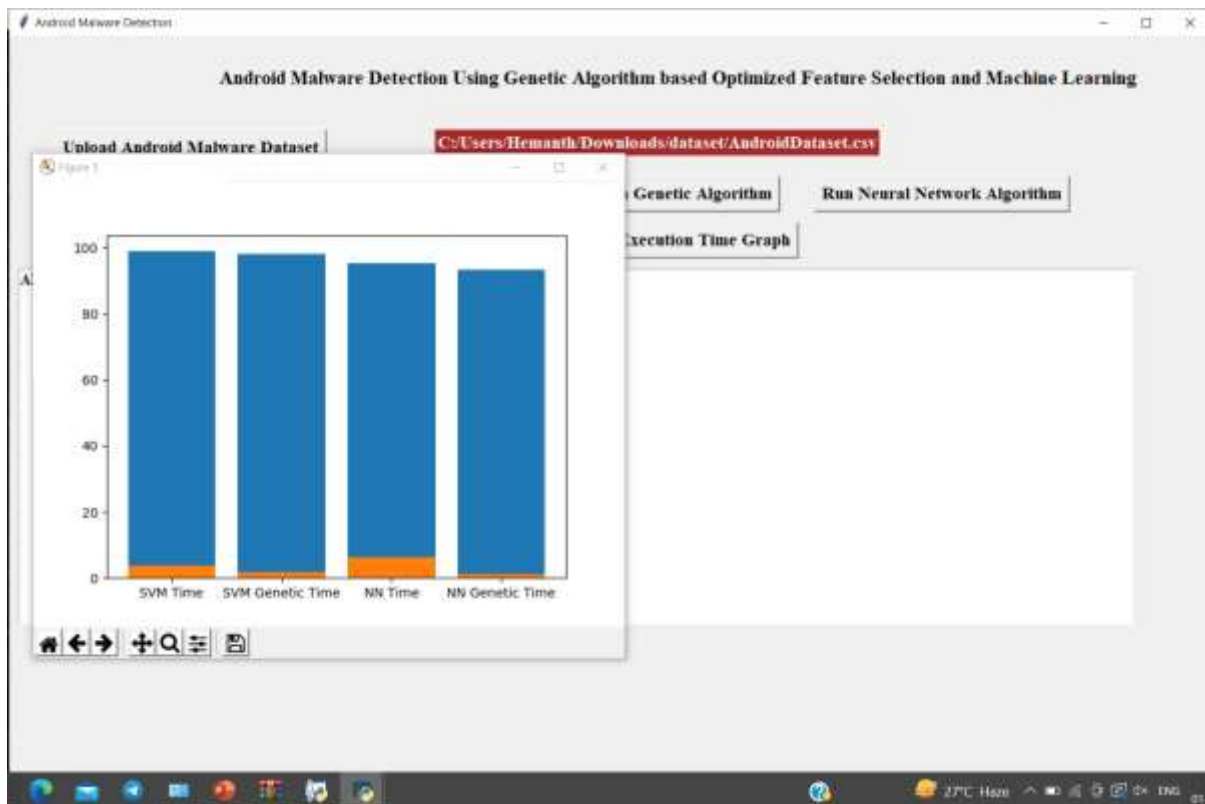


In above screen NN with genetic got 98.02% accuracy. Now click on 'Accuracy Graph' button to see all algorithms accuracy in graph





In above graph x-axis represents algorithm name and y-axis represents accuracy and in all SVM got high accuracy. Now click on 'Execution Time Graph' button to get execution time of all algorithm



In above graph x-axis represents algorithm name and y-axis represents execution time. From above graph we can conclude that with genetic algorithm machine learning algorithms taking less time to build model.

7. CONCLUSION

As the number of threats posed to Android platforms is increasing day today, spreading mainly through malicious applications or malwares, therefore it is very important to design a framework which can detect such malwares with accurate results. Where signature-based approach fails to detect new variants of malware posing zero-day threats, machine learning based approaches are being used. The proposed methodology attempts to make use of evolutionary Genetic Algorithm to get most optimized feature subset which can be used to train machine learning algorithms in most efficient way.

REFERENCES

- [1] D. Arp, M. Spreitzer, M. Hubner, 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.
- [2] N. Milosevic, A. Dehghantaha, and K. K. R. Choo, "Machine learning aided Android malware classification," *Comput. Electr. Eng.*, vol. 61, pp. 266–274, 2017.
- [3] J. Li, L. Sun, Q. Yan, Z. Li, W. Srisa-An, and H. Ye, "Significant Permission Identification for Machine-Learning-Based Android Malware Detection," *IEEE Trans. Ind. Informatics*, vol. 14, no. 7, pp. 3216–3225, 2018.



[4] A. Saracino, D. Sgandurra, G. Dini, and F. Martinelli, “MADAM: Effective and Efficient Behavior-based Android Malware Detection and Prevention,” *IEEE Trans. Dependable Secur. Comput.*, vol. 15, no. 1, pp. 83–97, 2018.

[5] S. Arshad, M. A. Shah, A. Wahid, A. Mehmood, H. Song, and H. Yu, “SAMADroid : A Novel 3-Level Hybrid Malware Detection Model for Android Operating System,” *IEEE Access*, vol. 6, pp. 4321–4339, 2018. shift in position,” *Biol. Cybern.*, vol. 36, no. 4, pp. 193–202, Apr. 1980.