WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Detecting Android Malware with an Enhanced Genetic Algorithm for Feature Selection and Machine Learning

B. Swarajya Lakshmi[1*], S. Pranavi[2], C. Jayalakshmi[3], K. Gayatri[4], M. Sireesha[5], A. Akhila[6]

[1]Assistant Professor in Department of Computer Science and Engineering, Santhiram Engineering College, Nandyal, Kurnool, Andhra Pradesh,India.
[2,3,4,5,6] Student,Department of Computer Science and Engineering, Santhiram Engineering College, Nandyal, Kurnool, Andhra Pradesh, India.
DOI: https://doi.org/10.55248/gengpi.2023.4.4.34984

ABSTRACT

Android's open source nature and Google's support have helped it garner the world's greatest market share. Being the most widely used OS in the world, it has attracted the focus of cybercriminals, who are active in a variety of ways but most notably via the widespread dissemination of malware software. In this research, we offer a machine-learning based method for Android malware detection that use an evolving Genetic algorithm to identify discriminating features. Machine learning classifiers are trained using the genetic algorithm's selected features and their ability to identify Malware is compared to its performance before and after feature selection. The experimental findings confirm that the genetic algorithm provides the best efficient feature subset, which aids in reducing the feature dimension to less than half of the original feature-set. For machine learning based classifiers, maintaining a classification accuracy of more than 94% after feature selection allows them to function on much decreased feature dimension, which has a beneficial effect on the computing cost of learning classifiers.

Keywords: SVM & ANN with Genetic Algorithm; Feature Selection

1. INTRODUCTION

The Google Play store is the official Android app store where users can browse and download apps for free. As a result of its free availability and due to Android's rising popularity among mobile device manufacturers and users, hackers have begun concentrating more and more on creating harmful programs for Android. Google Play store has taken a number of precautions to prevent malicious apps from reaching the general public, but some have still made it through. These apps can compromise users' privacy by collecting and sharing sensitive information, such as their contact lists, email addresses, and whereabouts, with unsavory parties. Since these malicious apps constitute a significant danger to Android systems, it is necessary to do malware analysis or reverse-engineer them. Both static and dynamic analyses may be performed on Android malware. Comparatively, dynamic analysis examines the runtime behavior of Android Apps in a controlled environment, whereas static analysis focuses on the code structure without actually running it. As the number of Android malware types that might pose zero-day risks continues to grow, it has become clear that a reliable system for detecting these threats is essential. In order to identify new varieties of Android Malware that pose zero-day risks, a machine- learning based technique combined with static and dynamic analysis is preferable to a signature-based strategy, which needs frequent updates of the signature database. The Support Vector Machine approach used for the comprehensive but lightweight static analysis presented in yielded respectable detection accuracy of 94%. Using a permissions-based approach and a representation of the source code as a bag of words, Nikola Milosevic demonstrated static analysis-based categorization. Another method based on determining the most important permissions and then using machine learning to assess the data is suggested in.With a detection accuracy of 96% and the ability to function exclusively on rooted devices, MADAM offers a multi-level analysis framework in which the behavior of Android apps is recorded up to four levels: from package, user, application, and kernel level. In order to increase the efficiency with which assets are used for mobile malware detection, SAM Droid introduced a revolutionary three-way host server based technique. The trend in malware detection now favors deep learning solutions with little human interaction. Feature selection is a critical process in any machine learning-based method. In addition to better experiment outcomes, avoiding the curse of dimensionality that plagues most machine learning based methods may be achieved by obtaining an ideal feature set. To boost detection accuracy generally, Fest suggested a new and effective approach for feature selection. For guidance in making this decision, see, which provides a summary of many feature selection techniques for malware detection. The suggested work utilizes the Genetic Algorithm for its ability to discover a feature subset from the original feature vector that provides the highest accuracy while training classifiers. It is used in tandem with machine learning and deep learning algorithms to find the best possible subset of features. The key contribution of this study is to use the Genetic Algorithm to decrease the feature dimension to less than half of the original feature-set so that it may be supplied as input to machine learning classifiers for training while preserving the accuracy of these classifiers in identifying malware. In place of the exhaustive feature-selection method—which involves evaluating 2N possible combinations, where N is the number of features—a heuristic searching strategy based on the fitness function known as the Genetic Algorithm has been applied. Two machine learning algorithms, the Support Vector Machine and the Neural Network, are trained using the genetic algorithm-optimized feature set. To reduce the training time complexity of classifiers, it has been discovered that a respectable classification accuracy of over 94% may be maintained despite working on a significantly smaller feature dimension. The remainder of the paper is laid out as follows:

2. RELATED WORK

"Drebin: Portable, Effective, and Easily Explainable Android Malware Detection" Insecure apps are a major problem for the Android operating system. Since their use is increasing and becoming more varied, traditional protections are becoming more ineffective. This leaves Android phones vulnerable to new forms of infection. Here, we offer DREBIN, a lightweight technique for detecting Android malware that allows for the identification of dangerous programs locally on the device. Since there are constraints on the resources that may be used to monitor programs in real time, DREBIN instead does a comprehensive static analysis, collecting as many characteristics as possible. The common patterns suggestive of malware may be automatically discovered and utilized to explain the judgments of our technique since they are contained in a shared vector space. DREBIN surpasses other similar techniques in an examination using 123,453 apps and 5,560 malware samples, detecting 94% of the malware with minimal false alarms, and the reasons supplied for each detection disclose key aspects of the discovered malware. The approach is excellent for analyzing downloaded programs immediately on the device, since an average examination on five common smart phones takes just 10 seconds. Cybercriminals have been using malware to launch assaults for decades. Since cell phones are used by so many people and can hold so much personal information, they have become a prime target for cybercriminals. It's no secret that Android is the most popular smart phone operating system.

"Machine learning supports Android malware categorization,"

Android is a tempting target for malicious coders, and the rapid proliferation of new malware strains and the prevalence of infected devices make manual malware analysis an impossible task. Researchers in the field of cyber forensics would be aided by the use of machine learning to the study of malware if they used it to uncover hidden patterns in the investigation of harmful software. In this research, we offer two machine learning-assisted strategies for static analysis of mobile applications: one based on permissions and the other on source code analysis using a bag of words representation model. In comparison to a strategy that relied just on permission names, our source code-based categorization scored 95.1% on the F-measure. Our technology shortens the time required to analyze smartphone malware and offers a means of automated static code analysis.

3. METHODOLOGY

There are two distinct categories of Android software, known as APKs: Reverse engineering is used by both malware and good ware to glean information about an app, including its permissions and the number of its many components. Providers of Content, etc. These characteristics are employed as a feature vector, and the Malware and Good ware classes are labelled as 0s and 1s in the CSV file.

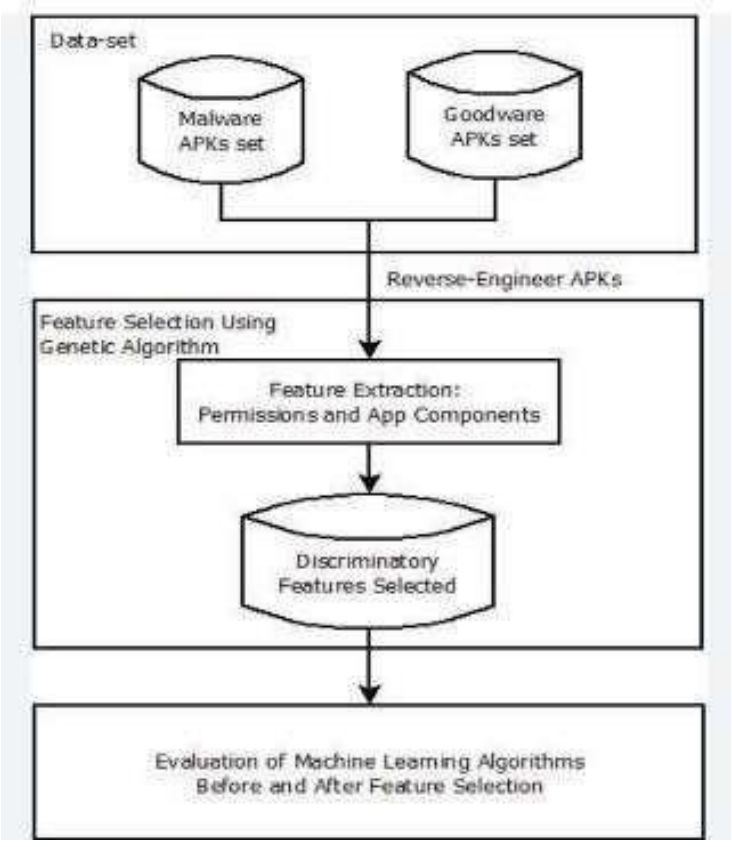

Figure: Methodology for Feature Selection Using Machine Learning

The CSV is then used as input into a Genetic Algorithm, which determines which characteristics are the most useful and eliminates some of the dimensions in the feature set. Once an optimum collection of features has been informing two different types of machine learning classifiers

4. RESULT AND DISCUSSION

Specifically, SVM (Support Vector Machine) and NN (Neural Network) are two machine learning methods that the author employs throughout this study (Neural Networks). It is necessary to optimize (lower the size of the dataset's columns) features since the app will have more than 100 of them, and building the model using machine learning would take too long. The author is doing so using a genetic algorithm. Using the dataset as a whole, the genetic algorithm will pick out the most relevant characteristics for use in training the model and discard the rest. This method will help minimize the amount of the dataset so that the training model may be built more quickly. After using the genetic algorithm, the accuracy drops somewhat, but the training time for the model is cut down significantly compared to the control group.

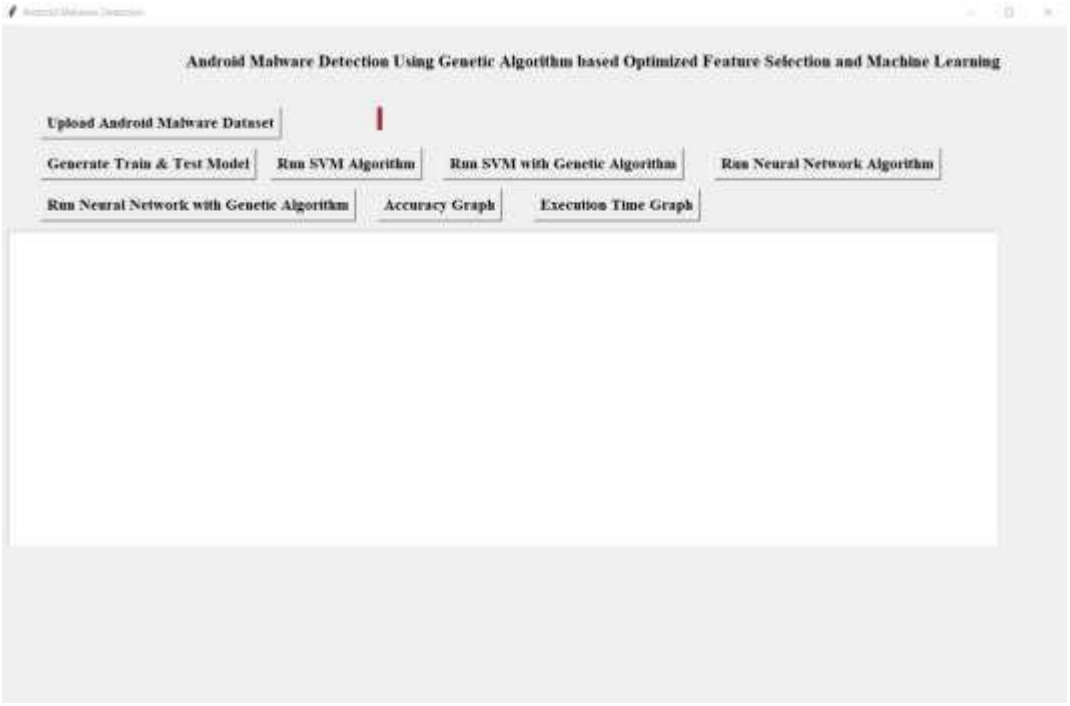

Fig: Output Screen

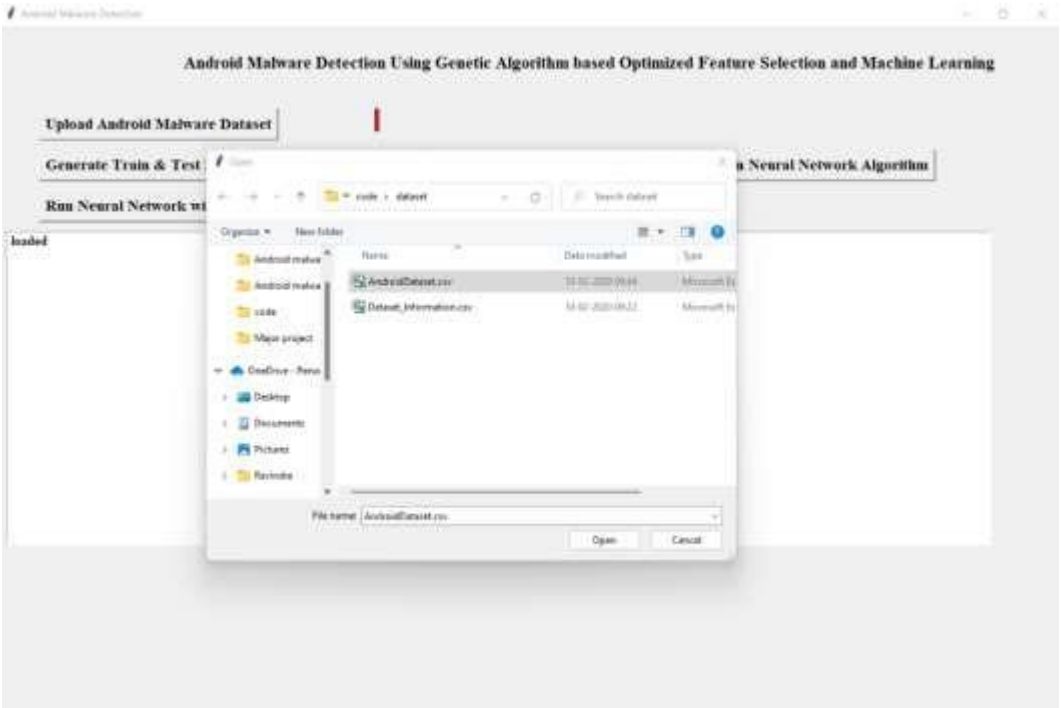

Fig: Upload dataset

Fig: Uploaded dataset

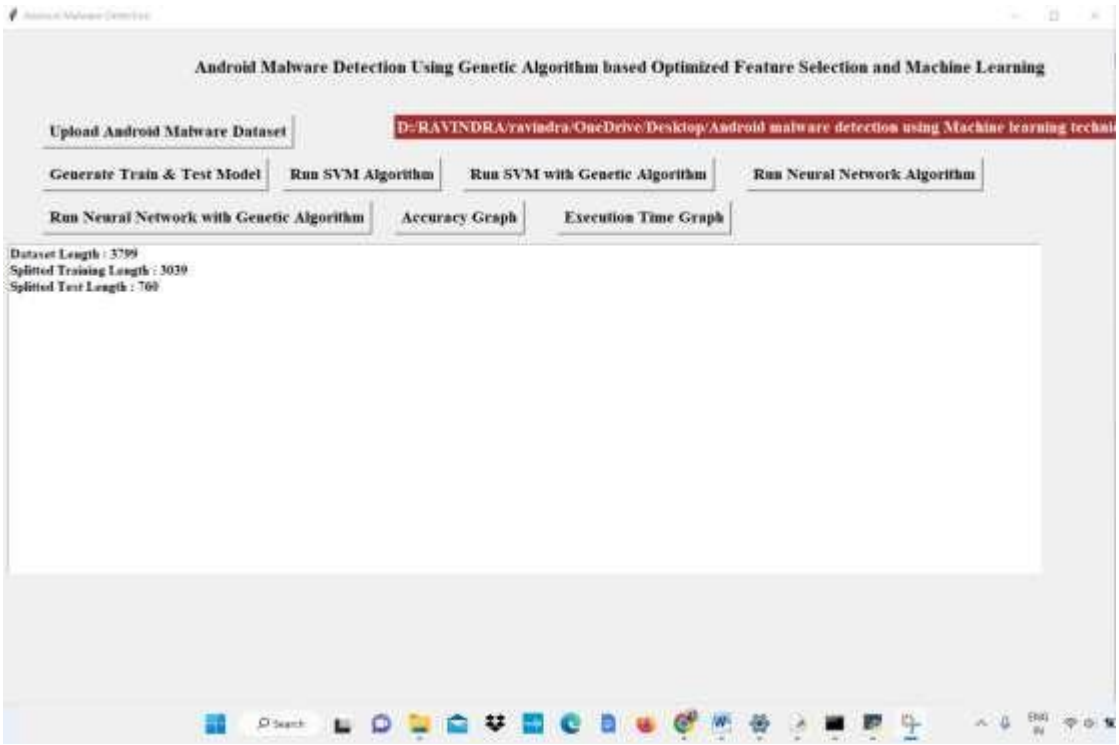

Fig: Generate Train &Test Model

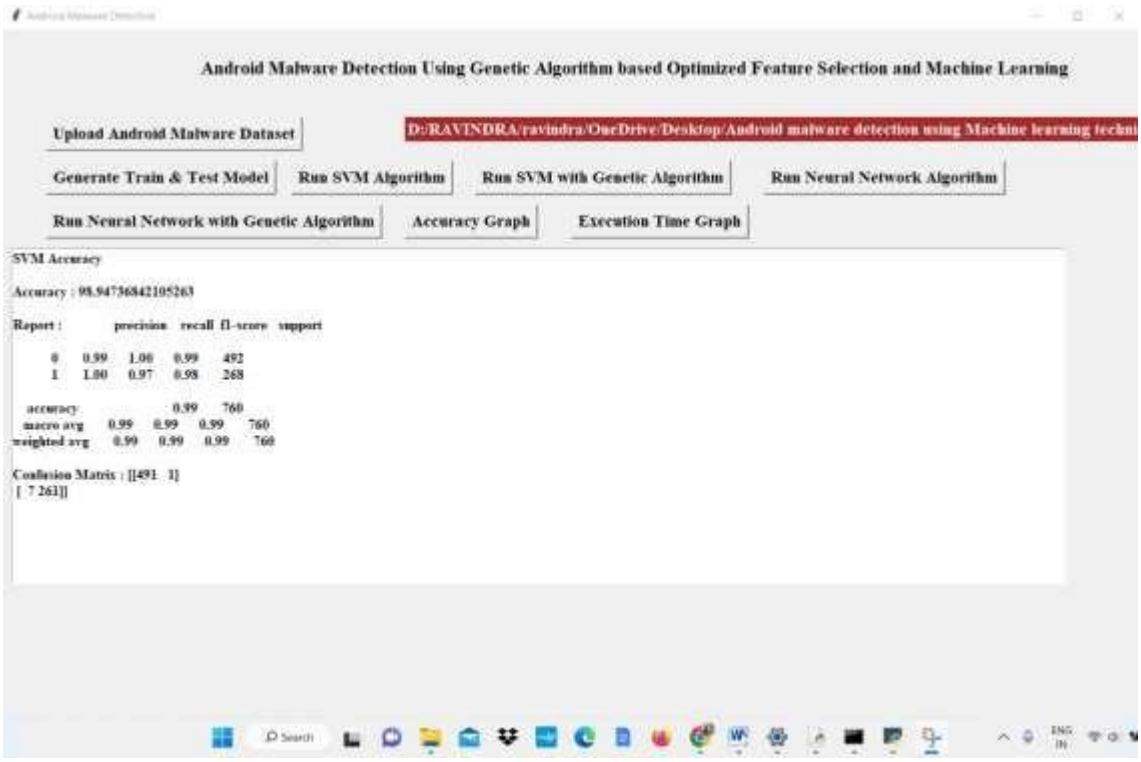

Fig: Run SVM Algorithm

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.

Fig: Run SVM with Genetic Algorithm

In above screen SVM with Genetic algorithm got 92% 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)

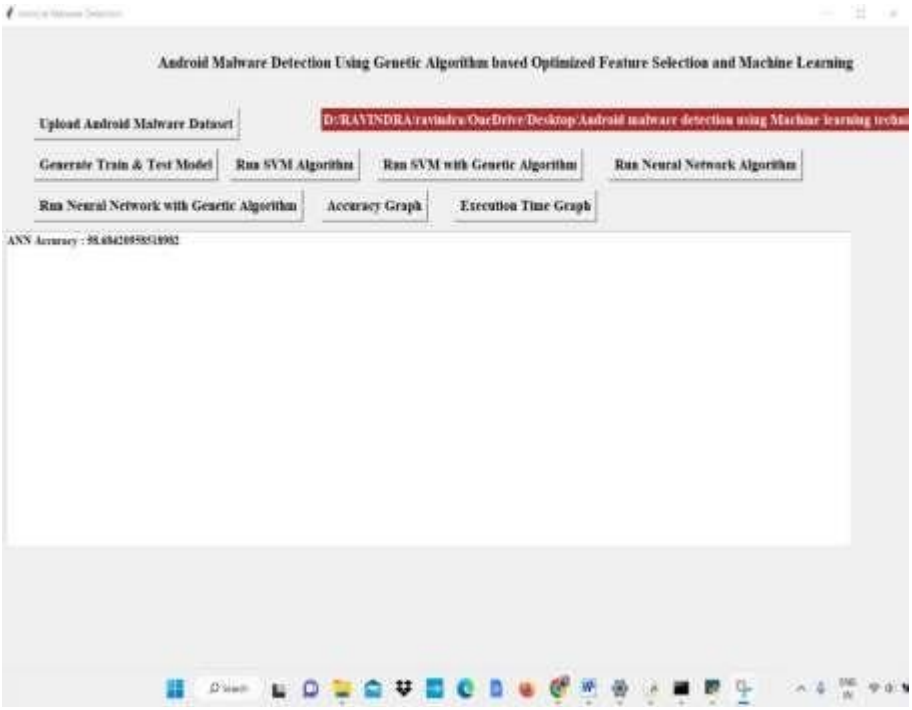

Fig: Run ANN Algorithm

In above screen neural network also gave 98.68% accuracy. Now click on 'Run Neural Network with Genetic Algorithm' button to get NN accuracy with genetic algorithm.

Fig: Run ANN with Genetic Algorithm

In above screen NN with genetic got 95.78% accuracy. Now click on 'Accuracy Graph' button to see all algorithms accuracy in graph.

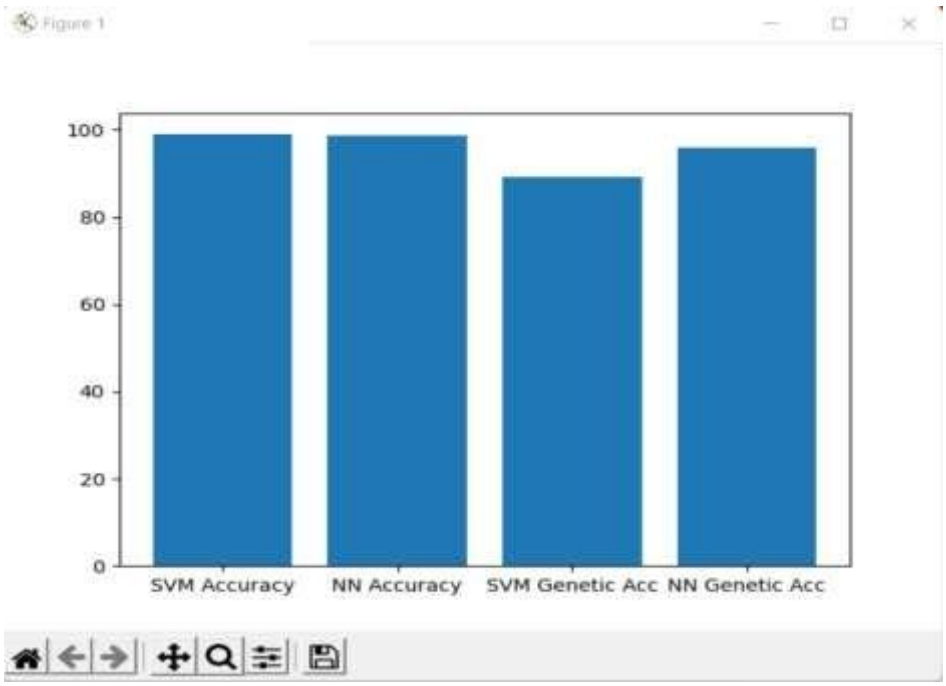

Fig: Accuracy Graph

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 algorithms.

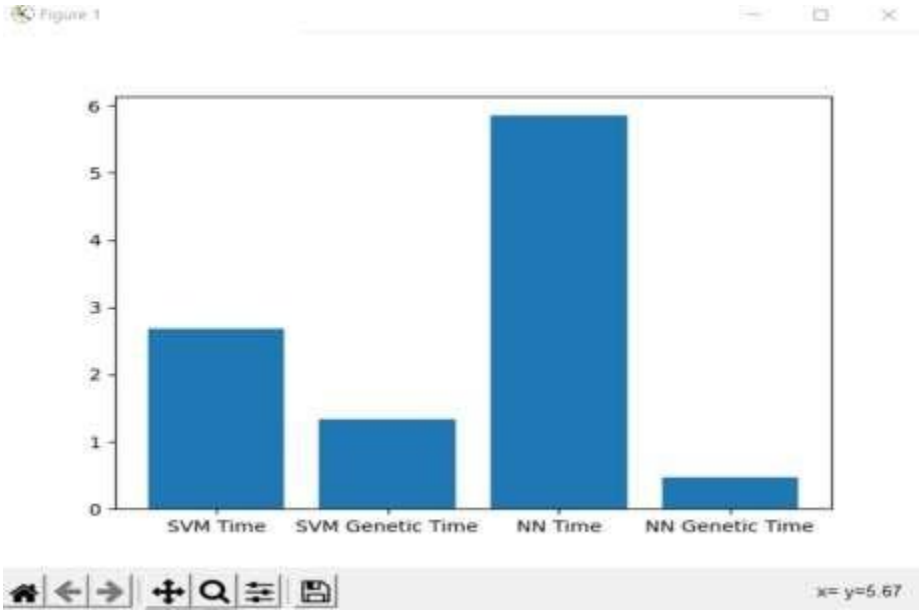

Fig: Execution Time Graph

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.

5. CONCLUSION

Malicious apps are a major source of the ever-increasing amount of Android security concerns; as a result, it is crucial that users take precautions. Conceive of a system that can accurately identify such malware and implement it. It is becoming common practice to apply machine learning based methodologies in cases when a signature based approach has failed to identify a new version of malware that poses a zero-day threat. The suggested technique employs an evolving Genetic Algorithm in an effort to discover the best possible collection of features for training machine learning algorithms. Experiments show that when working on a reduced dimension feature-set using Support Vector Machine and Neural Network classifiers, a respectable

classification accuracy of more than 94% is maintained. Using more extensive datasets and investigating the impact on other machine learning methods when combined with the Genetic Algorithm are two promising directions for future research.

6. REFERENCES

[1] B.Swarajya Lakshmi, " Fire detection using Image processing", Asian Journal of Computer Science and Technology ISSN: 2249-0701 Vol.10 No.2, 2021, pp.14-19, 2021.

[2] MV Subramanyam, "Automatic feature based image registration using SIFT algorithm", conference of 2012 Third International Conference on Computing, Communication and Networking Technologies (ICCCNT'12), pages. 1-5, July 2012.

[3] Sunar Mohammed Farooq, Nageswara Reddy Karukula, J David Sukeerthi Kumar,"A Study on Cryptographic Algorithm and Key Identification Using Genetic Algorithm for Parallel Architectures", International Advanced Research Journal in Science, Engineering and Technology ICRAESIT, Vol. 2, Special Issue 2, December 2015.

[4] Sunar Mohammed Farook, Jacob Jaya Raj, "An Efficient Layered Approach for Intrusion Detection System", International Journal of Computers Electrical and Advanced Communications Engineering, Vol.1 (3) ISSN: 2250-3129,2012.

[5] MV Subramanyam, Mahesh, "Feature based image registration using steerable filters and Harris algorithm", pages. 95-99, January 2012.

[6] Y Murali Mohan Babu, MV Subramanyam, MN Giriprasad , "Fusion and texture based classification of indianmicrowave data-A comparative study", International Journal of Applied Engineering Research, Vol. 10, no. 1,pages. 1003-1010, February 2015.

[7] M Sharmila Devi, Farooq Sunar Mahammad, D Bhavana, D Sukanya, TV Sai Thanusha, M Chandrakala, P Venkata Swathi "Machine Learning Based Classification and Clustering Analysis of Efficiency of Exercise Against Covid-19 Infection" , journal of algebraic statistics, Vol. 13, no. 3, pages. 112-117, June 2022.

[8] B.Swarajya Lakshmi, "Identity-Based Proxy-Oriented Data Uploading and Remote Data Integrity checking in Public Cloud", International Journal of Research Vol. 5, no.22,pages. 744-757, 2018.

[9] Bingi Manorama Devi, M Sharmila Devi, "Automatic classification and extraction of non-functional requirements from text files: a supervised learning approach", European Journal of Molecular & Clinical Medicine,Vol. 7,no.9,pages. 2231-2239, July 2021.

[10] V Lakshmi Chaitanya, "Machine Learning Based Predictive Model for Data Fusion Based Intruder Alert System", journal of algebraic statistics ,Vol. 13 ,no.2 ,pages. 2477-2483 ,June 2022.

[11] V Lakshmi Chaitanya, G Vijaya Bhaskar, "Apriori vs Genetic algorithms for Identifying Frequent Item Sets", International journal of Innovative Research &Development, Vol. 3,no. 6,pages. 249-254,June 2014.

[12] A. Martin, F. Fuentes-Hurtado, V. Naranjo, and D. Camacho, "Evolving Deep Neural Networks architectures for android malware classification", 2017 IEEE Congr. Evol. Comput. CEC 2017 - Proc., pp. 1659–1666, 2017.

[13] K. Zhao, D. Zhang, X. Su, and W. Li, " Fest: A Feature Extraction and Selection Tool for Android Malware Detection,", 2015 IEEE Symp. Comput. Commun., pp. 714–720, 4893.

[14] A. V. Phan, M. Le Nguyen, and L. T. Bui, " Feature weighting and SVM parameters optimization based on genetic algorithm for classification problems,", Appl. Intell., vol. 46, no. 2, pp. 455–469, 2017.