International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 06 Issue: 07 | July - 2022 Impact Factor: 7.185 ISSN: 2582-3930

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

GAMIDI NAGA VENKATA SIVARAMAKISHORE[1]

[1]*UG Scholar,Department of Electronics and communication Engineering,SRM university*

Abstract - The Android platform has the largest global market share because of its open-source nature and Google's support. Windows has grabbed the attention of fraudsters who exploit it to distribute malware since it is the most commonly used operating system on the planet. To produce the best-optimized feature subset that may be utilized to train machine learning algorithms most efficiently, the suggested methodology uses an evolutionary Genetic Algorithm for discriminatory feature selection. The ability of machine learning classifiers to detect malware before and after feature selection is examined. According to the outcomes of the experiments, the genetic algorithm gives the ideal feature subset, lowering the feature dimension to less than half of its original feature set. Machine learning-based classifiers maintain good classification accuracy despite working with lower and smaller feature dimensions, which reduces computational complexity.

Key Words: Genetic algorithm, Machine learning, Android malware, Reverse engineering

1. INTRODUCTION

Android Apps are freely accessible on Google Playstore, the official Android app store as well as third party app shops for consumers to download. Due to its open source nature and popularity, malware authors are increasingly concentrating on creating harmful apps for Android operating system. In spite of various attempts by Google Playstore 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 Playstore, the official Android app store as well as third-party app shops for consumers to download. Due to its open source nature and popularity, malware authors are increasingly concentrating on creating harmful apps for Android operating system. In spite of various attempts by Google Playstore 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 behavior 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.

2. PROBLEM STATEMENT

Because of the ubiquity of the Android operating system and the ease with which developers may create apps for it, anyone can create malware using ready-made tools. As a result, malware has spread throughout many helpful programmes, posing a risk to Android users. We have presented a method for detecting Android malware using permissions gained through static analysis. We choose significant features from the set of permissions using a combination of genetic algorithm and simulated annealing in the suggested method, and two algorithms, SVM and KNN, are created based on this approach. A dataset of 3799 samples with 1260 malware and 2539 benign programmes was used to test the suggested technique.The proposed method improves Android malware detection accuracy, and the SVM, Neural Network with genetic algorithm has the best result.

3.LITERATURE SURVEY

T. Kim .et.al.,[1] developed a new malware detection framework for Android. By analysing files like a manifesti file, a dex file, and a.so file from an APKi file, a total of seven types of features are retrieved, and these featuresi supplement the extractedi information to represent programme characteristicsi. The early neural network is trained using only certain types of features, and the results of the initiali networks are then used to train the final network.K. Zhao .et.al., [2] suggested a new and efficient feature selection approach to improve overall malware detection accuracy. They also released AppExtractor, a tool for extracting features from apps. They suggested FrequenSel as a solution to the absence of effective feature selection algorithms, compared it to other similar algorithms, and described its benefits and why it performs better.Sawle .et.al.,[3] This paper presents a comprehensive survey of machine learning-based Android malware detection approaches. He gave a quick overview of Android applications, including the Android system architecture, security procedures, and malware classification.
FIRDAUS, Badrul ANUAR, Ahmad KARIM, and Mohd Faizal Ab RAZAK[4] used Genetic Search(GS) to select the best features derived from permission, code-based, directory path, and system command categories.

4.METHODOLGY

There are two types of Android apps or APKs: Reverse engineering is used to extract features such as permissions and the count of App Components such as Activity, Services, and Content Providers. These properties are given as feature vectors in CSV format, with class labels of Malware and goodware represented by 0 and 1 respectively.
The CSV is sent to Genetic Algorithm, which selects the most optimised set of features to reduce the size of the feature set. Two machine learning classifiers, Svm and Nn, were trained using the optimized set of features collected.

The proposed methodology is depicted in Figure 1, which consists of two units: feature extraction using the Androguard tool and feature selection using the Genetic Algorithm. Finally, for evaluation, the selected features are supplied into machine learning algorithms.

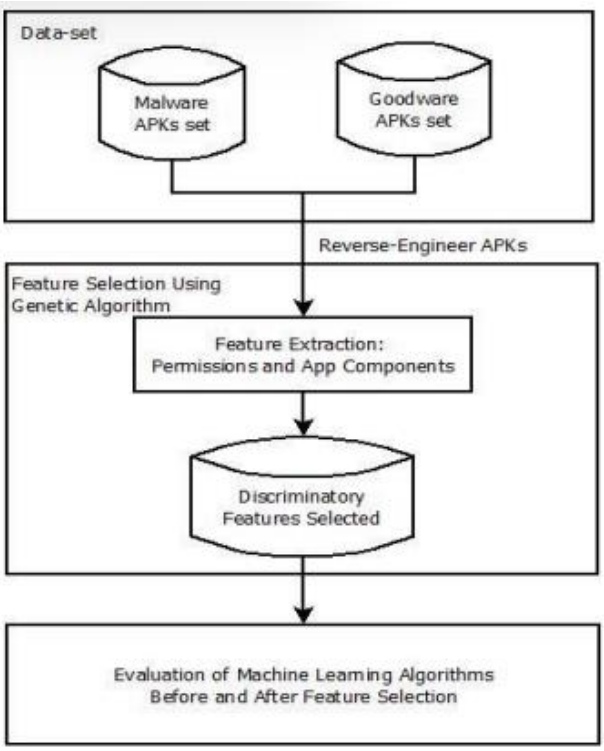


FIGURE -1

A. Reverse-Engineering of Android APKs

Static features are collected using the proposed methodology from AndroidManifest.xml, which provides all of the relevant information about the Apps required by any Android platform. The static features were obtained by disassembling the APKs with the Androguard tool.

B. Feature Vector

The following is how features are retrieved and mapped to a feature vector:

App Elements: A feature vector is created using the counts of App components like Activity, Services, Content Providers, and Video On demand.

Permissions: A |S| dimensional vector space is used to map the access-control feature set. There could be over 100 permissions in a single app (examples include transact, API call signature, nonservice-connected, API call signature, bindService, API call signature, attachInterface, API call signature, ServiceConnection, API call signature, android.os.Binder, API call signature, SEND SMS, Manifest Permission, Ljava.lang.Class.getCanonicalName, API call signature, and so on).We'll add value 1 in the features data if the app has necessary permission, and 0 if it doesn't. In this method, for each feature extracted from app x, a vector (x) is created with the relevant dimension set to 1 and all other dimensions set to 0.
It can be summarized in equation (1):
ψ: X {0;1}|S| → (1)

C. Discriminatory Feature Selection

Selecting the most critical features in malware detection is a crucial stage because it affects the quality of trial outcomes. Working with a low-dimensional feature vector including solely discriminatory features will also help to reduce the learning classifier's processing cost.
The dataset is fed through the Genetic algorithm, which produces the best subset of features for the machine learning-based classifier. The genetic algorithm keeps track of a population of features or chromosomes, as well as their fitness scores, such that chromosomes with higher fitness scores have a better chance of reproducing.The fitness function of a genetic algorithm is defined in such a way that the chromosome that gives the machine learning-based classifier high accuracy is given a higher value than features that give it lesser accuracy. Using the crossover and mutation process, the chromosomes with the best fitness score are chosen as parents to produce the next generation of kids.

The stages involved in utilising a Genetic Algorithm to pick features are listed below:

Step 1: Set up the method with feature subsets that are binary encoded, meaning that if a feature is included, it is represented by 1 in the chromosome and if it is not, it is represented by 0.

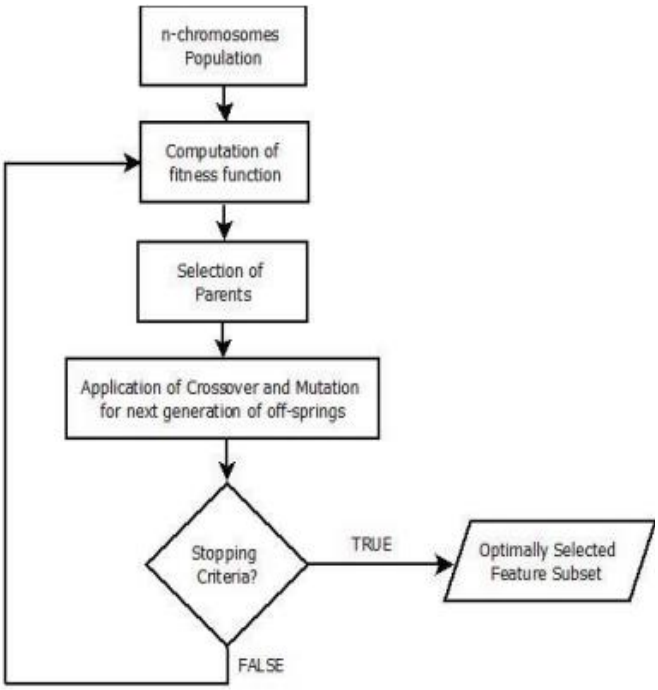
Step 2: Begin the procedure by establishing a randomly generated initial set of populations.

Step 3: Assign a fitness score to the genetic algorithm based on the defined fitness function.

Step 4: Picking Parents: To generate the next generation of offspring, chromosomes with high fitness scores are given preference over others.

Step 5: For the creation of offspring, perform crossover and mutation operations on the selected parents with the stated probability of crossover and mutation.

Iteratively repeat Steps 3 to 5 until convergence is achieved and the best chromosome from the population, i.e. the optimal feature subset, is acquired.

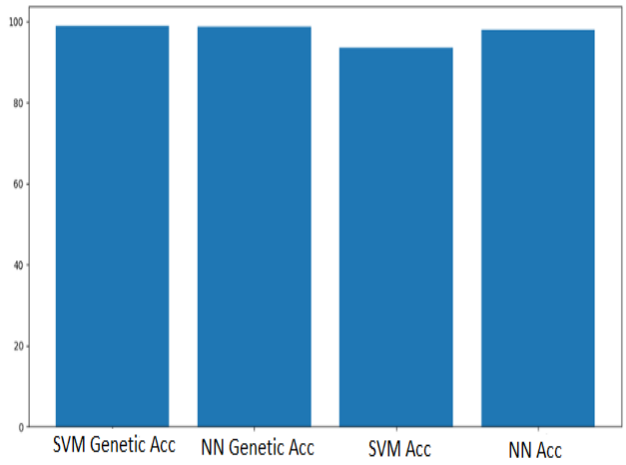


D. Machine Learning-Based Classification

Machine learning-based solutions are being chosen over a traditional signature-based strategy, which required regular updates of the signature database, due to the ever-increasing variations of Android Malware providing a zero-day threat. The Genetic Algorithm is used to pick features, which are then used to train and evaluate classifiers using the Support Vector Machine (SVM) and Neural Network techniques (NN).

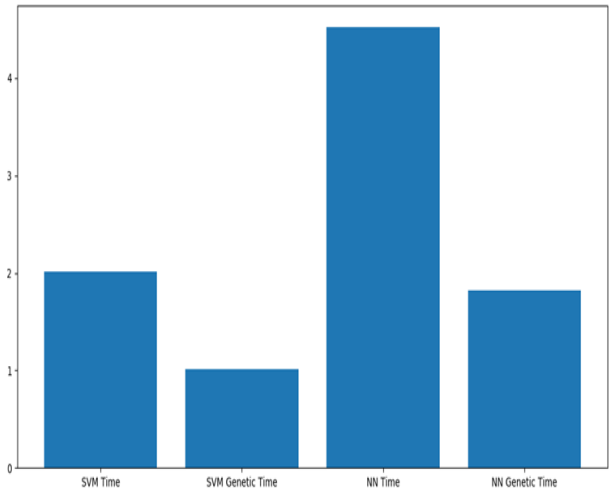
5.RESULT ANALYSIS

ACCURACY GRAPH:



In the above graph, the x-axis represents the algorithm name and the y-axis represents accuracy and in all SVM with genetic algorithm has high accuracy.

EXECUTION TIME GRAPH:



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

6. CONCLUSIONS

As the amount of dangers posed to Android systems grows every day, spreading primarily through malicious applications or malware, it is critical to develop a framework that can accurately detect such malware. Machine learning-based approaches are being employed where signature-based approaches fail to detect new types of malware posing zero-day threats. The proposed method seeks to employ the evolving Genetic Algorithm to obtain the best optimized feature subset that may be used to efficiently train machine learning algorithms. The results show that while dealing with a reduced dimension feature set and employing Support Vector Machine and Neural Network classifiers with genetic algorithm, respectable classification accuracy of more than 94 percent can be maintained while minimizing the training system complexity.

REFERENCES

1. Android Operating System, IEEE Access, vol. 6, pp. 4321–4339, 2018. [6] T. Kim, B. Kang, M. Rho, S. Sezer, and E. G. I'm, “A Multimodal Deep Learning Method for Android Malware Detection using Various Features,” vol. 6013, no. c, 2018
2, 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.

3.Sawle, P.D.; Gadicha, A. Analysis of malware detection techniques in android. Int. J. Comput. Sci. Mob. Comput. 2014, 3, 176–182.
4.Ahmad FIRDAUS, Nor Badrul ANUAR, Ahmad KARIM3,Mohd Faizal Ab RAZAK.Discovering optimal features using static analysis and a genetic search-based method for Android malware detection.2018