International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878, Volume-8 Issue-4, November 2019

Discover and Analyzes Whether Mobile Applications Downloaded From the Internet Are Good or Bad

G. Kalaimani, G. Kavitha,

Abstract: ***Android Malware is pernicious software. It is configured to attack the hardware such as android or mobile phone or smart phone. It is designed to exploit the flaw in specific mobile phone software technologies and operating systems. Nowadays, the mobile phone is the number one most vulnerable to malware attacks. Malware can be in the form of adware, Trojans, viruses, root kits and spyware. They delete important documents or steal protected data or bring software that is not authorized by the user. To solve this problem you need to categorize the applications on the mobile. The techniques used in machine learning are used here to differentiate between applications in mobile as good or bad. In this paper, present two methods as using the Genetic algorithm for feature selection and the Nearest Neighbor for classification.***

Keywords: Mobile malware, feature selection, classification, genetic algorithm, nearest neighbor.

I. INTRODUCTION

Nowadays, as the mobile phone is increasingly used by people, this malware is heavily vulnerable to attack. According to industry research, Apple strikes a more prominent attack on Android than the I Phone and OS. These malware includes stealing some types of data from the mobile and providing it to the hackers such kind of actions has taken place.

Worms and viruses of other types of malware work differently. Without the user's permission, the unknown person can perform certain actions that are contrary to their wishes from the user's mobile phone. This means installing applications that don't require to users, deleting some important information or stealing will do such unnecessary things. There are two other types of malicious software, such as spyware and adware. Spyware software steals important things like a user's identity card, credit card numbers. Adware software sends some unwanted ads.

We can prevent such functions by a different method. That is, we need to categorize new and unwanted applications that come into the user's mobiles as good or bad. The general user of such classification can know, and if bad, remove it immediately.

Manuscript published on November 30, 2019.
* Correspondence Author
Dr. G. Kalaimani, Professor*, Department of Computer Science and Engineering, shadan women's college of Engineering & Technology, Hyderabad, Telangana - 500004.
Dr. G. Kavitha, Professor, Department of Computer Science and Engineering, Muthayammal Engineering College, Kakkaveri, Rasipuram, Tamil Nadu 637408.

In [1], the author of this paper focuses on the idea that malware functions on mobile are increasing, that author has decided to prevent this. So from that point of view, they have chosen a deep learning system. However, this method is expensive and difficult to use. In [2], in this malicious software, security concerns were great problems.

Computer and mobile users, big corporations and governments are greatly affected by these. Algorithms of in deep and machine learning have been used. These are more time consuming.

In [3], they aim to detect malicious uses in android applications. Prior to that, utilities have extracted features and then used the boost method. But it does not effective to this problem. In [4], the features are first extracted and then categorized to detect maliciousness. Genetic algorithms have been used to extract features, and for classifications have been used in machine learning classifiers. But it is not easy.

In [5], the classification of applications is depending on the credence that malicious software can be determined, based on the confidence that, he proposed a machine learning system. But its accuracy was very low. In [6], in software ecosystems, certain types of malicious software have been developed to cause great harm. In this paper, methods such as SAE, SVM, cloud antivirus and NAE have been used to prevent this. More number of methods is used so require more storage space and more money.

In [7], they first analyze the malicious application and then identify it. They use behavioral analysis to analyze, and one of the machine learning methods for classification of malware. It was take more time. In [8], after first extracting the features, they find the application heard by the DOD in machine learning. It was never used that much. In [9], they use deep learning and hashing for security to detect malicious actions. There is no need to use the hashing function because the app is classified so it was security. In [10], artificial neural network system in machine learning is applied to detect harmful viruses in mobile. In their proposed method, the NGAV method was developed and found. It was not accurate and the cost was taken too much.

In [11], Some Neighbors steal the information that is most securely stored on mobile by malicious software. The authors of this paper have used cryptography methods to take this concept into account. But it is very difficult to detect this malicious software through this method. In [12], analyzing applications that first Contamination, It then extracted its features and selected the necessary ones and then categorized it. Genetic and SVM have used algorithms to perform these functions. It absorbs more energy.

II. PROPOSED METHEDOLOGY

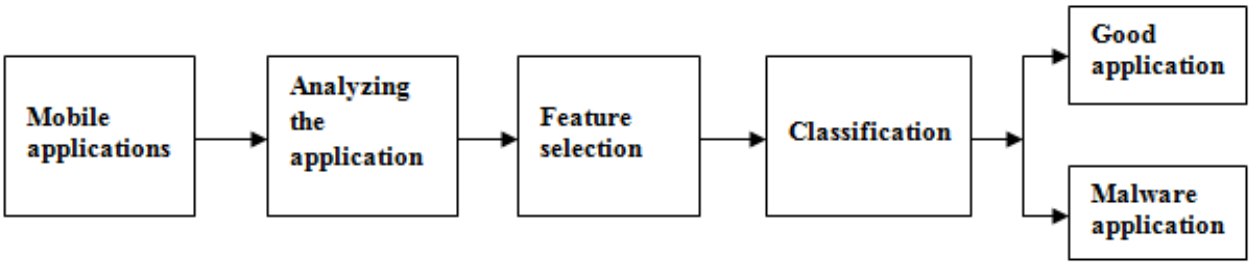

Fig.1: Flow of to detect the malware application in mobile

III. MALWARE ANALYSIS

Analyzing malware can collect information about malware. Doing so will help you develop tools to remove malware. Therefore, analyzing this malicious software can be very helpful in creating an Efficient and better tool. In the past, this analysis was done manually by various experts. This is a waste of time and a difficult task to find.

TYPES

There are many types of analysis:

1. Static analysis
2. Dynamic analysis
3. Threat analysis

1. STATIC ANALYSIS

Static analysis is the process of software debugging. This means that it directly analyze the malware. This static analysis helps the tools to determine if a file is intended to cause corruption or not.

2. DYNAMIC ANALYSIS

Dynamic analysis is the process of analyzing the malware's function, recognizing its functionality and learning its function. It is used in detection signatures when all kinds of details are obtained. This will identify the persons associated with the attacker. So it is with such actions as blackmail, and performing actions such as bringing the victim under control. These types of malware may be associated with dynamic analysis machines.

3. THREAT ANALYSIS

Threat analysis is the key to identifying malicious software. Finding information about Network's infrastructure is a difficult task. So hackers easily steal information from an individual's mobile phone. With this information they are intimidating the person involved and committing extortion.

IV. FEATURE SELECTION

Feature selection is the process of finding the necessary and relevant inputs for a prediction process. Therefore, these types of methods can be used for the accuracy of the forecasting process or delete unwanted features or for detecting the features required.

Mathematically stated, the method of selecting the inputs is very complex. This is caused by errors during data collection. In the neural network, input variables are addition (1) or omission (0). A complete solution of features will evaluate the various combinations. 2^N, N = Number of features. These types of activities require a lot of accounting work. Moreover, this is a very difficult task if the no. of features is high. So in real life, it requires easy methods to select features.

The most effective and efficient method of selection the feature is the genetic method. Through this genetic method can see how to select the most relevant features. Genetic algorithm is a heuristic selection method; it takes place through processes of natural evolution. This method helps to minimize the error in the model in the data set. This process is also called selection error, each of the information in use here application to the predictive model. Here, the genes is binary values and indicate whether or not to include a particular feature in the model. Every app inside the mobile has to select information about it. Now the functions of this algorithm can be found in detail.

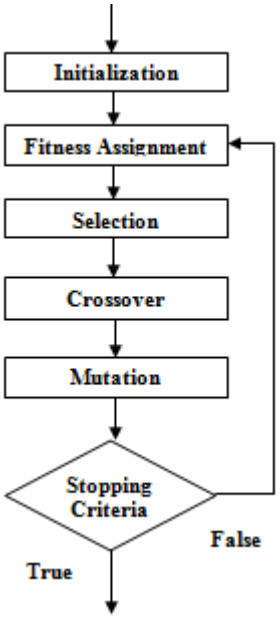

Fig.2: Flow chart for feature selection

1. INITIALIZATION

The first step is to create and launch data in applications. Because it is a random selection method, application's genes are usually initialized randomly. To illustrate this operator, there is a predictive model that is evaluated by a neural network with six features. If we create four application's information, there are four different neural networks with random features. The next image illustrates applications data.

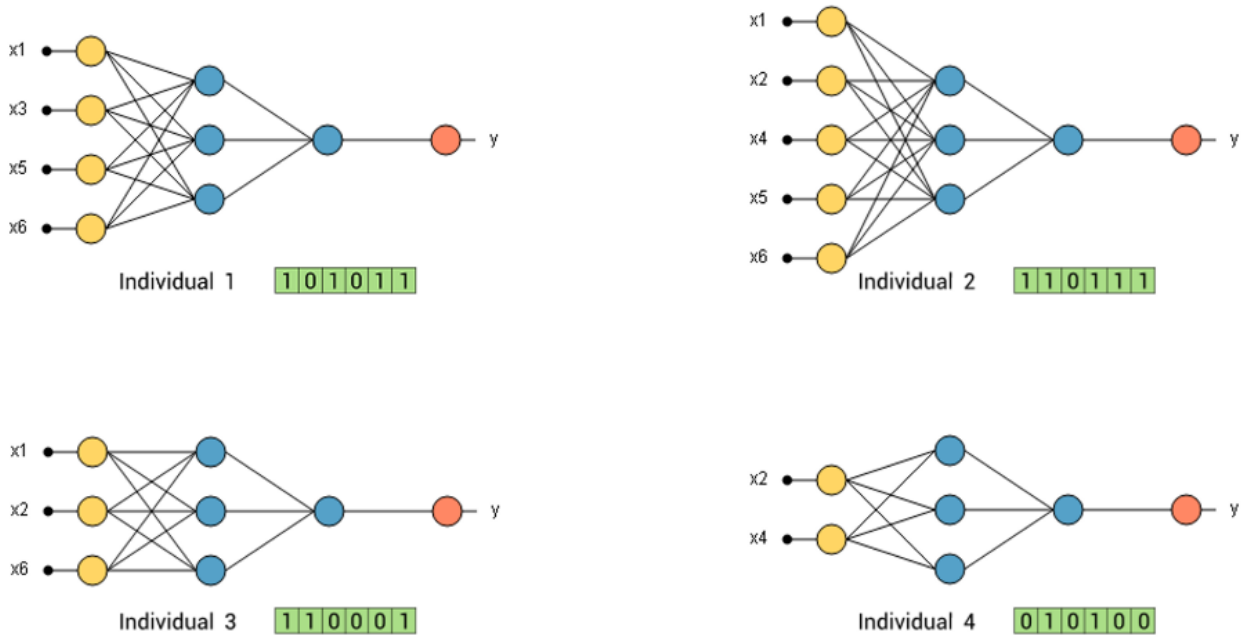

Fig.3: Initialization

Each application is represented in six binary genes. Each positive gene is, a related feature is included in the model.

2. FITNESS ASSIGNMENT

Once the app.'s has collected the information, to each application must give qualify? To rate the fitness, the predictive model should be trained with training data. Then its selection error should rate with selection data. If the choice error is high, it means that the exercise is low. There should be a higher probability of being selected to reunite with higher fitness applications. The most quota used method of exercise allocation is rank-based. With this method, all applications are sorted by selection errors. Then, the exercise allocated to each application depends on its position in the individual applications status.

The rate of the exercise assigned to each application with the rank-based method is given below:

$$\Phi (i) = k * R (i)$$

Where,

i = 1,…,N

k = selective pressure

k's value is adjusted between one and two. Highly selected stress values have a high probability of combining the best applications.

R (i) = the quality of the individual e.

3. SELECTION

Upon completion of the exercise, the selection operator selects the individual applications according to the exercise. The no. of applications designated is introduced to as N/2. N = the no. of applications. The size of the height is also controlled by the number of nicely selected applications.

4. CROSSOVER

Once the selection operator has selected applications, the crossover operator re-connects the selected applications and creates new applications. In this step, Select two applications at random and must integrate their features into the new application, still As long as the new application is the same size as the old one. The crossover method helps determine whether or not each feature comes from the application.

5. MUTATION

The Crossover operator can generate very similar information for applications. Some aspects of mutation operator applications solve this problem by randomly changing the value. Create an alignment between Zero and One to find out if a feature will change. This is called the mutation rate. The mutation rate is chosen as 1/m, where, m = Number of features, Changing the features of each application with values for the mutation rate.

V. CLASSIFICATION

Identifying and removing unknown malware that contains useless forms is a difficult task. The KNN system is one of the most common and basic procedure in machine learning. So we utilize the KNN mechanism to resolve this complication that is mobile applications. It is a mechanism of owning a supervised learning purpose. It is a method of owning a supervised learning method. Greatly used in intrusion detection, data processing and authentication. It is not parameterized, so it is widely dispersed in real life scenarios. That is, it does not perform basic functions of data distribution. Here are giving some pre-rendered data's application. It classifies applications into groups identified by an attribute.

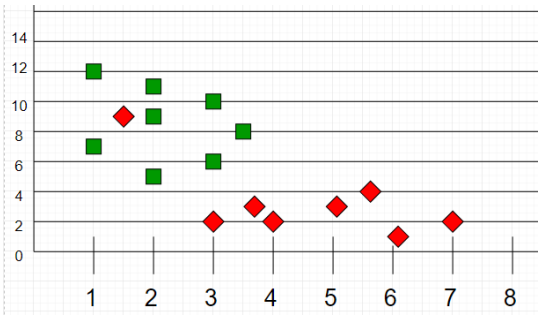

Fig.4: Data points of application

Now if you give the data points of another application, by analyzing the training set can assign these points as a group. In fig.5, the unclassified points are called white.

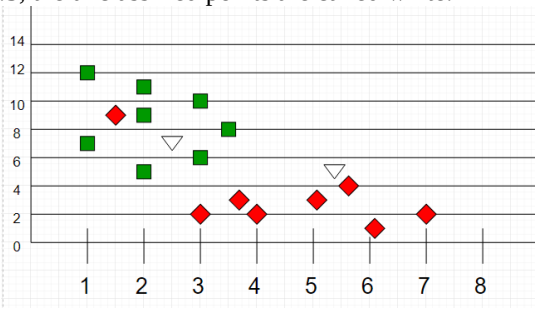

Fig.5: Unclassified Points

1. INTUITION

If you draw these points on a graph, some groups or clusters can be found. Now if you give an unclassified point, it should be noted that the neighbors belong to which set, then allocate it to a set. This c pointed that the closest point to the cluster of points classified as red is classified as red, because its probability is high. Intuitively, you can see in Fig.5, that the first point (2.5, 7) is classified as green and the second point as (5.5, 4.5) is red.

ALGORITHM

Take m is the no. of data representative and p is the unknown point.

1. Save the training samples in an array of data points arr [], this means that every element in the array represents a tuple (x, y).
2. For i = 0 to m:
   a. Calculate the Euclidean distance d (arr [i], p).
3. Make K's S a very small distance, each of these distances Together with an already classified data point.
4. Turn the foremost label in the middle S.

VI. RESULT AND DISCUSSION

This paragraph gives a rundown of the results of the project and judgment on it. The result of the first method is to choose the features in the mobile applications; we used the Genetic algorithm in this project for that purpose.

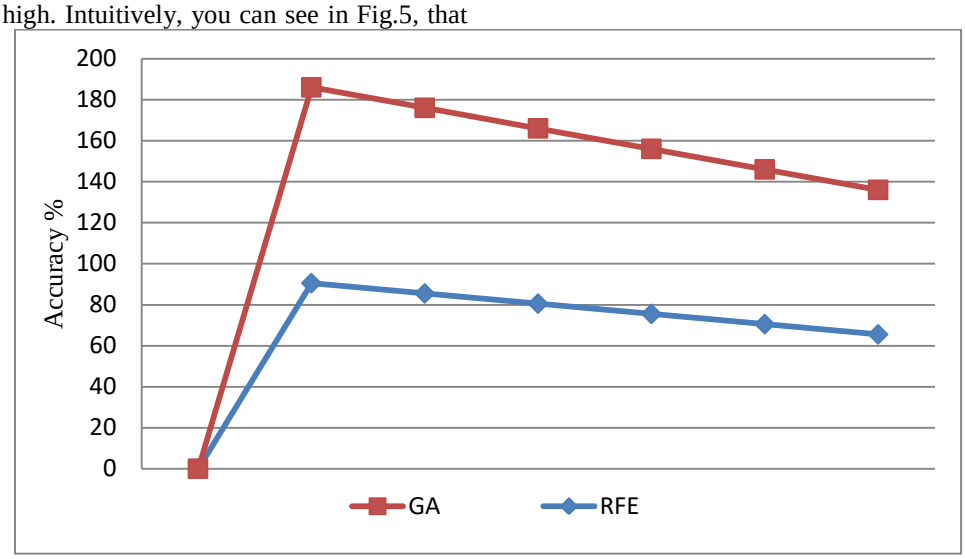

Fig.6: Performance of the accuracy of the GA mechanism

The cause is that it gives a result that is more efficient, accurate, quicker and cost effective than other systems. Likewise, this method has given the expected results. This means that in Fig.1 genetic algorithm is much more efficient than the RFE system.

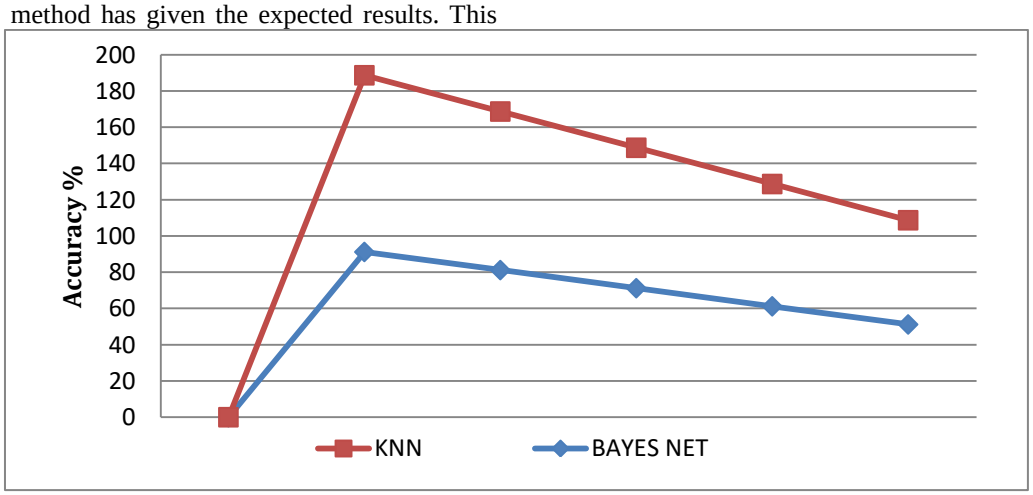

Fig.7: Accuracy Performance of the KNN method

The second result is that the selected features are assigned to the classifier as input. Here we use the KNN method as an assortment. This classifier classifies android applications as good and bad. In that respect, this classification is showing in fig.7 to prove better than the Bayes Net classification in accuracy.

VII. CONCLUSION

When looking at this project, it seems that feature selection and classification is very important for finding a bad application. Each data set has a number of features. So choose the most useful ones. To do this, one needs an advanced, more efficient method, that method is the genetic method. This is much better than the traditional feature choice. Next step is classification, the KNN procedure is utilized in this step. This is because it is very easy and easy to understand or explain or accurate. Much better compared it to other learning methods. And it can be utilized for regression or categorization. Both methods have therefore yielded good results. Finally using these methods the malware application is more accurately detected.

REFERENCES

1. Josh McGiff, William G. Hatcher, James Nguyen, Wei Yu, Erik Blasch, and Chao Lu, "Towards Multimodal Learning for Android Malware Detection", International conference on computing, networking and communications(ICNC): Communications and Information Security Symposium, @ 2019.
2. Vinayakumar R, Mamoun Alazab, Prabaharan Poornachandran, and Italakshmi Venkatraman, "Robust Intelligent Malware Detection Using Deep Learning", IEEE. Translations and content mining are permitted for academic research only, DOI: 10.1109/Access.2019.2906934, IEEE Access.
3. Qi Fang, Xiaohui Yang, Ce Ji, "A Hybrid Detection Method for Android Malware", IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC 2019), 978-1-5386-6243-4/19/$31.00 © 2019 IEEE.
4. Anam Fatima, Ritesh Maurya, Malay Kishore Dutta, Radim Burget and Jan Masek, "Android Malware Detection Using Genetic Algorithm based Optimized Feature Selection and Machine Learning", Department of Telecommunications, Brno University of Technology, Brno, Czech Republic, 978-1-7281-1864-2/19/$31.00 ©2019 IEEE.
5. Xiaoqin Fu, Haipeng Cai, "On the Deterioration of Learning-Based Malware Detectors for Android", IEEE/ACM 41st International Conference on Software Engineering: Companion Proceedings (ICSECompanion), 2574-1934/19/$31.00 ©2019 IEEE, DOI 10.1109/ICSE-Companion.2019.00110.
6. Anav Bedi, Nitin Pandey, Sunil Kumar Khatri, "Analysis of Detection and Prevention of Malware in Cloud Computing Environment", Amity Institute of Information Technology, 978-1-5386-9346-9/19/$31.00 ©2019 IEEE.
7. Matus Uchnar, Peter Fecilak, "Behavioral malware analysis algorithm comparison", SAMI 2019 IEEE 17th World Symposium on Applied Machine Intelligence and Informatics, January 24–26, Herl'any, Slovakia, 978-1-7281-0250-4/19/$31.00 ©2019 IEEE.
8. Jordan Pattee and Byeong Kil Lee, "Implications for Hardware Acceleration of Malware Detection", 2019 IEEE 30th International Conference on Application-Specific Systems, Architectures and Processors (ASAP), 2160-052X/19/$31.00 ©2019 IEEE DOI 10.1109/ASAP.2019.00-14.
9. Sean Park, Iqbal Gondal, Joarder Kamruzzaman, Jon Oliver, "Generative Malware Outbreak Detection", 978-1-5386-6376-9/19/$31.00 ©2019 IEEE.
10. Ricardo Pinheiro, Sidney Lima, Sergio Femandes, Edison Albuquerque, "Next Generation Antivirus Applied To Jar Malware Detection Based On Runtime Behaviors Using Neural Networks", Proceedings Of The 2019 IEEE 23rd International Conference On Computer Supported Cooperative Work In Design, 978-1-7281-0350-1/19/$31.00 ©2019 IEEE.
11. Junggab Son, Euiseong Ko, Uday Bhaskar Boyanapalli, Donghyun Kim, Youngsoon Kim, and Mingon Kang, "Fast and Accurate Machine Learning-based Malware Detection via RC4 Ciphertext Analysis", 2019 Workshop on Computing, Networking and Communications (CNC), 978-1-5386-9223-3/19/$31.00 ©2019 IEEE.
12. Areeba Irshad, Ritesh Maurya, Malay Kishore Dutta, Radim Burget, Vaclav Uher, "Feature Optimization for Run Time Analysis of Malware in Windows Operating System using Machine Learning Approach", Department of Telecommunications, Brno University of Technology, 978-1-7281-1864-2/19/$31.00 ©2019 IEEE.