IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Improving Android Malware Detection With Keyword Vectors And Support Vector Machines

1st Syed Abrar Azhar,
ME Student,
Computer Science & Engineering,
Everest College of Engineering and Technology,Aurangabad(MH), India,

2nd Dr. V. S. Karwande,
Assistant Professor,
Computer Science & Engineering,
Everest College of Engineering and Technology,Aurangabad(MH), India,

Abstract— The usage of mobile phones has increased significantly all over the world as a result of the introduction of the internet and the passage of time. When it comes to mobile operating systems, Android is the one that is used on the most devices. There has been a significant increase in the amount of malicious applications that are designed to target mobile devices as a result of the widespread availability of smartphones. The discovery of new forms of malicious software and versions of harmful software has gotten increasingly difficult in recent years. Some Keywords for Our Method Through the use of correlation distance, it is possible to correlate important codes that are contained within Android malware source code. API calls, Android permissions, common arguments, and phrases that are regularly used are all included in these crucial codes. As a result, the Support Vector Machine (SVM) is utilized, which makes it possible to recognize both newly developed malicious software as well as malware samples that have already been constructed. In contrast to more conventional ways, this strategy is distinguished by the fact that it is not associated with the context of the text. The actions of dangerous programs are documented through the use of this approach, which involves the integration of an operating system with many forms of malicious software.

Keywords: Android, malware, the correlation distance between keywords, and the super vector machine (SVM).

I. INTRODUCTION

This analysis indicates a substantial rise in the quantity of harmful applications for Android. The importance of smartphones has markedly risen in recent years within the framework of our daily life. It is unequivocal that Android is currently the most prevalent smartphone operating system. This proclivity has led to a growing number of harmful programs being accessible in both official and unauthorized Android markets. "Malware" is an acronym for the term "malicious software." It is, in fact, harmful software that was clandestinely installed on the computer without the owner's knowledge during the installation procedure. It can be employed to gain access to computer systems or to gather data of paramount significance. A malware infection can result in several consequences, from mere user irritation to the theft of personal information. Moreover, there has been a surge in the popularity of smartphones based on Android platforms, especially those exhibiting superior performance. Malware detection solutions must be developed to ensure a secure environment for Android, given the substantial rise in dangerous applications in recent years. To compare each program against the database of known malware signatures, conventional content signatures, including a compilation of malware signature definitions, are employed. They offer an approach utilizing feature extraction based on the keywords vector. A detrimental attack can be executed using a singular set of preset statements. They are very aware that just a little fraction of the requests they receive is likely to do harm. It is customary to execute a succession of malevolent actions to inflict damage. Binary programs are the basis of conventional feature extraction techniques; yet, these methods also offer a strategy grounded in the correlation distance between words. The technology they utilize for identifying new types of malware and versions of harmful software is SVM-based feature vector identification.

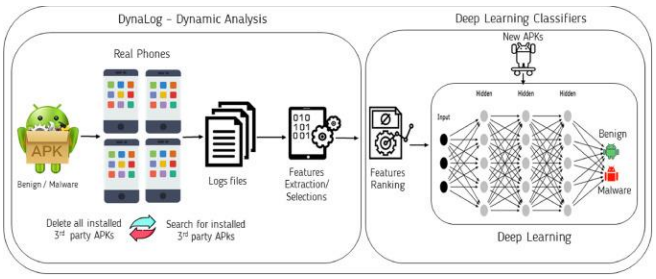

Figure 1: Android Malware Detection.

II. LITERATURE SURVEY

Threats of malware to computer systems, Android-based smartphones, and Internet of Things (IoT)-based systems have emerged alongside the proliferation of online services and smart device connection. Protecting the system's resources, data, and information from malware assaults is a crucial function of anti-malware software. In order to conceal their harmful operations, malware programmers nowadays employ sophisticated techniques like as obfuscation, packing, encoding, and encryption. Conventional malware detection

systems are rendered useless in the face of these sophisticated malware evasion methods. Many researchers have previously been interested in cyber security and have worked on malware detection models based on Machine Learning (ML) and Deep Learning (DL). An extensive literature review on malware detection methodologies is included in this paper. Reviews of feature selection (FS) methods suggested for malware detection, reviews of ML methods suggested for malware detection, and reviews of DL methods suggested for malware detection make up the bulk of the malware detection literature. After reviewing the existing literature, we have pinpointed the areas where further study is needed, as well as potential future directions for developing a more effective framework for detecting and identifying malware.[1]

The majority of smart gadgets are now under the process of having the Android operating system loaded. There is a significant increase in the number of incursions that are being introduced into operating systems of this kind. As a result of the introduction of such dangerous data streams, smart devices are now vulnerable to a wide variety of assaults, including phishing, spyware, SMS fraud, bots, banking Trojans, and many more. In this work, the implementation of machine learning classification techniques for the purpose of ensuring the safety of Android APK files is considered possible. On the basis of a variety of factors, every single apk data stream was classified as either malicious or the absence of harmful activity. After that, the techniques of machine learning classification are utilised in order to determine whether the signature of the newly installed programs belongs to the domain of malicious or non-malicious applications. If it is determined that it is harmful, then the right measures may be taken, and the Android operating system can be protected from behaviours that are considered to be prohibited.[2]

In the realm of internet, malware refers to harmful software that has been purposefully designed and continues to constitute a significant risk. In recent years, Android malware has become one of the most important hazards that can be found on the internet due to the fact that its frequency has increased. Despite the fact that a significant amount of effort has been put into detecting and classifying malware for Android, there are still issues that have not been fixed. These issues include data theft, difficulty in recognising malware for Android, and detection that takes a lot of time. This study suggested a Deep Learning (DL) Based Malware Attack Detector in Android Smartphones utilising LinkNET (MADRAS-NET) that efficiently identifies and mitigates the various forms of malware that are present in Android devices. The purpose of this investigation was to find a solution to this problem. In order to begin the pre-processing procedure, the Max Abs Scaler is initially given a collection of data as input. The result is then submitted to LinkNET for categorisation when the pre-processing step has been completed. Following the identification of malware by LinkNET, the output is then segmented into three categories: actual users, Penetho malware, and FakeAV malware. LinkNET uses the pre-processed data to detect malware. Last but not least, the MADRAS-NET technique that was provided is evaluated with the help of the AndMal2020 dataset, which allows for the detection and classification of malware as well as families of malware. The proposed LinkNET achieves a maximum accuracy of 99.81%, whereas other models, such as the Deep Belief Network (DBN) with a 96.75% accuracy rate, the Generative Adversarial Network (GAN) with a 94.42% accuracy rate, and the Long Short Term Memory Network (LSTM) with a 93.11% accuracy rate, obtain results that are comparable. A comparison of the MADRAS-NET model with the LSTM, GAN, and DBN frameworks reveals that the MADRAS-NET model has superior performance characteristics.[3]

The challenge of detecting and classifying harmful software (also known as malware) is a difficult undertaking, and there is no technique that is completely effective. A great deal of work remains to be done at this point. When it comes to malware detection, consistent benchmarks are tough to come by, in contrast to the majority of other study topics. Deep learning (DL) in text and image classification, the use of pre-trained and multi-task learning models for malware detection approaches to obtain high accuracy, and which approach is the best if we have a standard benchmark dataset are some of the topics that will be covered in this paper. The purpose of this paper is to investigate recent advancements in malware detection on MacOS, Windows, iOS, Android, and Linux using deep learning (DL). We review the effectiveness of these DL classifiers and their inability to explain their decisions and actions to DL developers, presenting the necessity of using Explainable Machine Learning (XAI) or Interpretable Machine Learning (IML) programs. This allows us to discuss the issues and challenges that arise when attempting to detect malware using DL classifiers. In addition, we address the influence that adversarial assaults have on deep learning models, which has a detrimental impact on their generalisation skills and leads to poor performance on data that has not been observed before. We are of the opinion that it is necessary to utilise various malware datasets in order to train and evaluate the efficacy and efficiency of the deep learning models that are now considered to be state-of-the-art.On a variety of datasets, we investigate eight well-known deep learning algorithms. Researchers will be able to establish a broad grasp of malware identification capabilities using deep learning with the assistance of this survey.[4]

As a result of the growing prevalence of malicious software for Android smartphones, mobile devices are confronted with major security concerns. By integrating Support Vector Regression (SVR) with dynamic feature analysis, this research presents a novel method for detecting malware on Android devices. The goal of this strategy is to solve the growing number of difficulties that are associated with mobile security. The purpose of our study was to design a malware detection system that is more accurate and dependable, and that is also capable of recognising many varieties of malware, both known and unknown. A thorough approach was established by us, which included the extraction of dynamic features from Android applications, the preprocessing and normalisation of features, and the utilisation of SVR in conjunction with a Radial Basis Function (RBF) kernel for the classification of malware. The SVR-based model achieved a 95.74% accuracy, 94.76% precision, 98.06% recall, and a 96.38% F1-score, beating benchmark algorithms such as SVM, Random Forest, and CNN. Our findings illustrate the better performance of the SVR-based model. Through the use of ROC analysis, the model demonstrated an outstanding capacity for discrimination, achieving an Area Under the Curve (AUC) value of 0.98.The capability of the proposed model to capture intricate, non-linear connections in the feature space considerably boosted its efficacy in discriminating between applications that were not harmful and those that were malicious.In addition to providing academics and security practitioners with useful insights that can be used to solve developing malware concerns, this research lays a solid basis for the advancement of Android malware detection systems.[5]

A growing number of harmful software programs are being developed on a daily basis in tandem with the growing use of smart gadgets. The Android operating system is the one that is most often utilised in smart devices. As a result, this platform is the target of a significant amount of malicious software. It is possible to identify whether or not a piece of software is harmful by looking at the permission attributes that it possesses. However, this is a difficult issue to deal with. In this study, classification procedures have been carried out in order to identify whether or not the program is hazardous using machine learning techniques. This was done in order to find a satisfactory solution to the problem. In order to accomplish this goal, a dataset was developed that included 2854 pieces of malicious software and 2870 pieces of safe software. There are 116 permission features for each piece of software included in the dataset, as well as a class feature that indicates whether or not the software is dangerous. The Support Vector Machine (SVM) and Naïve Bayes (NB) models were trained with the help of these features. During the training and testing phases, the 10-fold cross validation approach was utilised successfully. Metrics like as accuracy, F-1 Score, precision, recall, and specificity were utilised in order to conduct an analysis of the different models' performances. We utilised the ROC curve and the area under the curve (AUC) values to conduct an analysis of the learning and prediction levels of the models. As a consequence of the tests that were conducted on the models, the SVM model had a classification success rate of 90.9%, while the NB model achieved a success rate of 92.4%.[6]

There is a significant portion of the mobile terminal market that is occupied by the Android operating system. On the other hand, it encourages the quick creation of applications (apps) for Android. The development of malicious software for Android, on the other hand, poses a significant threat to the safety of Android smartphone users. A great number of approaches for detecting malware on Android have been provided by previous research works; however, these methods did not make use of the information on the functional category of the app. As a result, the high resemblance that exists between innocuous apps that fall under the same functional category was not taken into consideration. The purpose of this work is to present a malicious software detection method for Android that is based on the functional categorisation. The benign applications that fall within the same functional category are more comparable to one another. As a result, we may utilise fewer features to identify malicious software, which allows us to enhance the detection accuracy within the same functional category. We intend to do this by developing a technique of functional categorisation that is capable of automatic application and has a high degree of precision. The hyperlink induced topic search (HITS) technique serves as the inspiration for the functional classification approach that we build for implementation in Android applications. Using the outcomes of the automatic categorisation, we proceed to create a method for detecting malware that is based on the similarity of applications that fall under the same functional category. For the purpose of evaluating our strategy, we make use of both healthy applications from the Google Play Store and malicious applications from the Drebin malware collection. Based on the findings of the experiments, it is clear that our approach has the potential to significantly enhance the precision of malware identification.[7]

Over the course of the previous several years, there has been a rise in the quantity of malicious software for Android. The statistics from VirusShare, which demonstrates that the quantity of malware is growing with each passing year, lends credence to the assertion that this is the case. Because of this, it is essential to categorise the malware in order to identify the many forms of malware that are infecting smartphones, which will ultimately make it simpler to find a solution to the problem. In order to solve the problem, this research categorises malicious software for Android, according to the forms of malware. Activities, permission, and receiver are the properties that are utilised, and they are all found inside the Androidmanifest.xml file. These findings are derived from the VirusShare database in 2018, which include information about malware. Support Vector Machine (SVM) in conjunction with RBF kernel and k-fold equal to 10 were the classification techniques that were utilised. In addition, gain ratio feature selection was utilised in this investigation in order to reduce the number of superfluous features that were included in the data. The classification obtained through the use of feature selection has a 72.5% accuracy rate. In comparison to the classification result that did not use feature selection and had an accuracy of 72.8%, this value was 0.3 in the other direction. On the other hand, the data that is categorised through the use of feature selection has the potential to cut the process of creating a classification model by 206 seconds.[8]

As a result of the widespread adoption of smart phones over the course of the past decade, it would appear that these electronic devices have evolved into an essential component of day-to-day living. People in this day and age are using a variety of new and different technologies, and they have a tendency to save vital data on their mobile phones. The unfortunate reality is that data related with privacy is the primary focus of attention for cybercriminals. As a result, hackers in this day and age are employing new methods to intercept and store the information that is stored on the mobile devices of users. Through the utilisation of pre-existing anti-virus softwares, which are able to identify applications that often contain a known and existing virus type. On the other hand, the fresh new sort of unknown variety is not detectable in a typical situation. This article presents a technique that is based on artificial neural networks (ANN) and support vector machines (SVM) to identify malicious applications and applications that are not malicious. Collecting and aggregating features of the necessary permissions and features that have been utilised as a list is the notion that is now being realised. When it comes to classifying the unknown applications, we implement it by utilising SVM and ANN. In the experimental findings, it was found that the Support Vector Machine (SVM) had an accuracy rate of 93%, while the Artificial Neural Network (ANN) had an accuracy rate of 90.89%. This means that the SVM was able to correctly identify both benign and malicious applications, even for those that were unknown.[9]

It is the most recent smartphone operating system that has been utilised all over the world, and it now has around 80 percent of the market share. A total of 3.48 million applications are currently accessible for download from the Google Play Store. The rapid increase in the number of dangerous applications available in the Google Play Store and other third-party app stores has, regrettably, become a major cause for concern, which in turn slows down the development of the Android smartphone ecosystem. A new harmful application is released every ten seconds, according to a survey that was conducted not too long ago. The malicious applications that are being developed are designed to carry out a wide range of threats, including viruses, worms, Trojan horses, and exploits. In order to solve this problem, a novel method that is both efficient and effective for detecting

malware in Android applications has been developed. This method makes use of the Aquila optimiser and the Hybrid LSTM-SVM discriminator. Methods: In this paper, the optimal features are selected from the CSV file based on the prediction accuracy by cross validation using the Aquila optimiser. The mean square error (MSE) obtained by the cross validation is considered to be the fitness function for the Aquila to select the optimal features through the use of the cross validation. In order to forecast the sort of malware that is present in the android system, the results indicate that the best features that were extracted are sent to the Hybrid LSTM-SVM classifier for the purposes of training and testing the features. The conclusion is that this suggested model is implemented on Python 3.8 for performance measures such as accuracy, precision, execution time, error, and other similar metrics. When compared to other methods, such as LSTM, SVM, RF, and NB, the suggested model can achieve an accuracy of 97%, which is higher than the accuracy achieved by other approaches. As a result, the suggested methodology automatically forecasts the presence of malicious software within the Android application.[10]

For mobile devices like smartphones and tablets, Android has reached between 70 and 80 percent of the market share. Consequently, cybercriminals have moved their destructive operations to mobile platforms in tandem with the rise in popularity of mobile platforms. Threat posed by mobile devices Researchers have identified a concerning rise in the danger posed by malicious software for Android devices between the years 2012 and 2013. They have calculated that the number of harmful programs that have been found is anywhere between 120,000 and 718,000. Numerous attempts have been made to investigate the characteristics of smart phone platforms and the programs that run on them, with the goal of effectively detecting malware from applications that originate from the workplace and from third-party sources. Through the implementation of machine learning strategies, the ultimate objective of this study is to enhance the permission to identify dangerous Android mobile applications. The first step is to collect a dataset consisting of prior harmful applications to use as a training set. Then, with the assistance of the Support Vector Machine algorithm and the Decision Tree algorithm, a comparison is done between the training dataset and the trained dataset. As a result of this, we are able to forecast the malware and mobile applications for Android that are roughly up to 93.2% of the time unknown or new malware. SIGPID, which stands for Significant Permission Identification, is a system that is implemented in this article in development. SIGPID is utilised to increase the effectiveness and efficiency of the permissions granted to programs, as well as to boost the accuracy of the detection of malicious software applications. An analysis of the differences between the training dataset and the trained dataset is carried out in this study by employing machine learning methods such as support vector machines (SVM) and decision tree algorithms. The methods that are employed in support vector machines serve as a classifier that is utilised to differentiate between malicious applications and benign applications. Confusion matrix with support vector machine (SVM), Confusion matrix with decision tree, and Confusion matrix Naïve Bayes are the methods that are utilised to implement this job. Using a confusion matrix in conjunction with a decision tree, it has been discovered that the greatest number of malicious apps have been identified.[11]

Significantly more people throughout the world are using smartphones, which has led to a rise in the number of assaults that are directed against these devices. Although several security strategies for detecting malware on Android have been offered, the majority of these strategies do not have the capability to identify malware at an early stage. In light of this, there is an urgent requirement to develop a system that can detect dangerous applications prior to making use of the data. Additionally, attaining a high level of accuracy in the detection of malware traffic on Android devices is another significant challenge. The purpose of this study is to present a deep learning system that can identify malware on Android devices by using network traffic properties. In order to function properly, machine learning algorithms often require data preparation; nevertheless, the processes of preprocessing are time-consuming. With deep learning approaches, there is no longer a requirement for data pretreatment, and these techniques are effective in addressing malware detection issues. Using the one-dimensional convolutional neural network (CNN), we extract local features from network flows. Then, we use the long short-term memory (LSTM) algorithm to identify the sequential link between the significant features. To identify malicious software for Android, we make use of a real-world dataset called CICAndMal2017 that contains information related to network traffic. In the binary, category, and family classifications scenarios, our model obtains an accuracy of 99.79 percent, 98.90 percent, and 97.29%, respectively.[12]

III. Proposed Methodology

It is possible to see a schematic depiction of the system architecture in the figure that can be seen above. As opposed to the traditional binary programming technique, our system employs a feature extraction method that is based on the correlation distance between keywords. This is in contrast to the ordinary binary programming strategy. Another technique that is distinct from the traditional binary programming approach is this one. Within the framework of this approach, the extraction of Java code from an APK file as well as the extraction of keywords are both activities that are carried out. Additionally, it serves the goal of checking the permissions that are contained within the Android manifest file. In addition, we make use of feature vectors in order to precisely define the characteristics of potentially harmful software. Not only do these features consist of application programming interfaces (APIs), but they also comprise similar parameters, common packages, and other components that are equivalent to one another. Based on our findings, we have determined that we provide a solution for detecting malware that is founded on support vector machines (SVM) and the feature vector set technology. It is possible to discover new malware and risky software versions using this strategy, which is able to do so while retaining an approach that is still very basic.

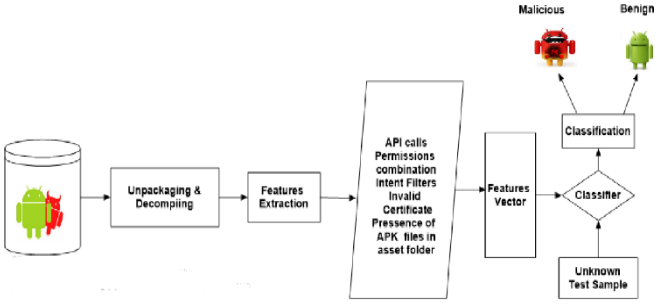

Figure 3.1: System Architecture

For the purpose of detecting malware on Android devices, a KCD-based extraction approach was added into the system that was suggested, and it was successfully deployed. Following this, the feature is combined with other characteristics to produce a keywords feature vector, which is then utilised in the succeeding process. Finally, support vector machines (SVM) are utilised to not only learn information but also to make judgements in order to recognise new types of dangerous software and variants that are hazardous. There are a number of ways in which this system is different from the normative approaches that are often utilised. It has been demonstrated through experiments that the technique is both effective and efficient in identifying malicious software on Android-based computers. Through the use of several experimental procedures, this has been demonstrated.

IV. EXPERIMENTAL RESULTS

A KCD-based extraction technique for Android malware detection was incorporated in the system that was suggested, and it was successful in being deployed. Next, the feature is joined with other features to create a keywords feature vector, which is subsequently used. At last, SVM is used to acquire knowledge and make judgements in order to identify new forms of malicious software and variations that are hazardous. There are a number of ways in which this system is distinct from conventional methods. Experiments have shown that the technology is effective and efficient in recognising malware on Android systems. This has been shown via the processes of experimentation.

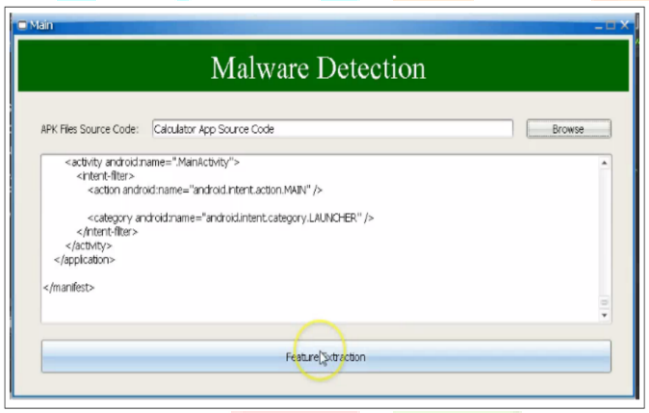

Figure 4.1: Feature Extraction

V. CONCLUSION

A KCD-based extraction technique for Android malware detection was incorporated into this system, and it ultimately proved to be successful. Next, the feature is joined with other features to create a keywords feature vector, which is subsequently used. At last, SVM is used to acquire knowledge and make judgements in order to identify new forms of malicious software and variations that are hazardous. There are a number of ways in which this system is distinct from conventional methods. Experiments have shown that the technology is effective and efficient in recognising malware on Android systems. This has been shown via the processes of experimentation.

REFERENCES

1. Santosh K. Smmarwar,Govind P. Gupta and Sanjay Kumar,"Android malware detection and identification frameworks by leveraging the machine and deep learning techniques: A comprehensive review",Telematics and Informatics Reports, 12 March 2024.
2. Madiha Tahreem,Ifrah Andleeb and Arsalan Hameed,"Machine Learning-Based Android Intrusion Detection System",Intrusion Detection System,December 6, 2024.
3. Yi Wang and Shanshan Jia,"MADRAS-NET: A deep learning approach for detecting and classifying android malware using Linknet ",Elsevier,11 March 2024.
4. Ahmed Bensaoud,Jugal Kalita,Mahmoud BenSaoud,"A survey of malware detection using deep learning,Machine Learning with Applications 16,Elsevier,2024.
5. Nahier Aldhafferi,"Android Malware Detection Using Support Vector Regression for Dynamic Feature Analysis",MDPI,19 October 2024.
6. Abdullah Batuhan Yilmaz,Yavuz Selim Taspinar,Murat Koklu,"Classification of Malicious Android Applications Using Naive Bayes and Support Vector Machine Algorithms",IJISAE,2022.
7. Wenhao FAN,Dong LIU,Fan WU,Bihua TANG, and Yuan'an LIU,"Android Malware Detection Based on Functional Classification",IEICE Trans. Inf. And Syst.,Vol.E105–D, No.3,March 2022.
8. Hendra Saputra,Amalia Zahra,"Classification of Android Malware Types Using Support Vector Machine",JTAIT,Vol.100. No 5,15th March 2022.
9. Prasanna Kumar G,Prasanna N Bhat,Prathik B.P.,Sachin Mirji,Prof. Poornima M,"Machine Learning Approach to Learn and Detect Malware in Android",IRJMETS,Volume 04,Issue:07,July-2022.
10. M. Gracea and Dr. M. Sughasiny,"Malware detection for Android application using Aquila optimizer and Hybrid LSTM-SVM classifier",EAI Endorsed Transactions on Scalable Information Systems,22 August 2022.
11. Dr. D. Hema Latha,Dr. D. Rama Krishna Reddy,Sudha Katkuri,"Prediction of Malicious Android Applications Using Machine Learning Approaches",IJRPR,Vol. 2, no. 10, pp. 430-436, October 2021.
12. Mahshid Gohari, Sattar Hashemi and Lida Abdi,"Android Malware Detection and Classification Based on Network Traffic Using Deep Learning",ICWR,IEEE,2021.
13. Rohit Srivastava,R.P. Mishra, Vivek Kumar, Himanshu Kumar Shukla,Neha Goyal and Chandrabhan Singh,"Android Malware Detection Amid COVID-19",IEEE Conference,December, 2020.
14. Talal A.A Abdullah,Waleed Ali and Rawad Abdulghafor,"Empirical Study on Intelligent Android Malware Detection based on Supervised Machine Learning",IJACSA,Vol. 11, No. 4, 2020.
15. Jiaqi Pang and Jiali Bian,"Android Malware Detection Based on Naive Bayes",IEEE,2019.

16. Vrushal R. Patil,Prof. Shital Jadhav,"A Review on Malware Detection on Android Smartphones Using Keywords Vector and SVM",IJIRSET,Vol. 7, Issue 1, January 2018.
17. Md Shohel Rana,Andrew H. Sung,"Malware Analysis on Android Using Supervised Machine Learning Techniques",IJCCE,Volume 7, Number 4, October 2018.
18. Sergii Lysenko,Kira Bobrovnikova,"SVM-based Technique for Mobile Malware Detection",IEEE,2018.
19. Junmei Sun,Kai Yan,Xuejiao Liu,Chunlei Yang,Yaoyin Fu,"Malware Detection on Android Smartphones using Keywords Vector and SVM",IEEE,2017.
20. Anshul Arora, Sateesh K Peddoju and Mauro Conti,"PermPair: Android Malware Detection using Permission Pairs",IEEE,VOL. 14, NO. 8, AUGUST 2015.