Tuijin Jishu/Journal of Propulsion Technology
ISSN: 1001-4055
Vol.45 No. 1 (2024)

Investigation on Challenges and Modern Implications for Android Malware Detection Using Machine Learning

Dr. Pokkuluri Kiran Sree[1], Gowtham A.[2], Fathima H.[3], T Yogameera[4], Dr. D.Shanthi[5], R.Revathi[6]

[1]*Professor & Head, Department of Computer Science and Engineering, Shri Vishnu Engineering College for Women, Bhimvaram, Andhra Pradesh, India – 534202,*

[2]*Assistant Professor, Department of Computer Science and Engineering (Cyber Security), Madanapalle Institute of Technology & Science, Andhra Pradesh, India – 517325,*

[3]*Assistant Professor, Department of Computer Application (BCA), K.S.Rangasamy College of Arts and Science (Autonomous), Tiruchengode, TamilNadu, India – 637215,*

[4]*Assistant Professor, Department of Computer Science and Engineering, Theni Kammavar Sangam College of Technology, TamilNadu, India - 625534,*

[5]*Associate Professor & Head, Department of Computer Science and Engineering, PSNA College of Engineering and Technology, Dindigul, TamilNadu, India- 624622,*

[6]*Assistant Professor, Department of Computer Science and Engineering, Theni Kammavar Sangam College of Technology, TamilNadu, India - 625534,*

Abstract:- Malware or malicious software is a term for various viruses, spyware, or ransom ware that cause harm to a user, their data, or their devices. Android operating system ranks first in the market share due to the system's smooth handling and many other features that it provides to Android users, which has attracted cyber criminals. Traditional Android malware detection methods, such as signature-based methods or methods monitoring battery consumption, may fail to detect recent malware. The term malware implies malicious intent on the side of the software developer. In recent times very sophisticated and complicated malware are being produced on a regular basis and it has grown into one of the stealthiest and lethal attack techniques used against critical information technology infrastructures. During 2022, the worldwide number of malware attacks reached 5.5 billion, an increase of two percent compared to the preceding year, according to Kaspersky, 80.69% of attacks on mobile users belonged to malware. This paper provides a systematic review of ML-based Android malware detection techniques, enables researchers to acquire in-depth knowledge in the field and to identify potential future research development directions and to understand the current situation with android apps and also provides a few approaches to develop future technologies for android systems.

Keywords*: Android, Malware Detection, Smartphone, Regression, Neural Networks, Classification, Android security, malware detection, code vulnerability, machine learning.*

1. Introduction

Malware can infect networks and devices and is designed to harm those devices, networks and/or their users in some way. Depending on the type of malware and its goal, this harm may present itself differently to the user or endpoint. In some cases, the effect malware has is relatively mild and benign, and in others, it can be disastrous. For example, malicious programs can be delivered to a system with a USB drive, through popular collaboration

tools and by drive-by downloads, which automatically download malicious programs to systems without the user's approval or knowledge.

In this technological era, smartphone usage and its associated applications are rapidly increasing due to the convenience and efficiency in various applications and the growing improvement in the hardware and software on smart devices. It is predicted that there will be 4.3 billion smartphone users by 2023. Android is the most widely used mobile operating system (OS). As of May 2021, its market share was 72.2%. The second highest market share of 26.99% is owned by Apple iOS, while the rest of the 0.81% is shared among Samsung, KaiOS, and other small vendors. Google Play is the official app store for Android-based devices. The number of apps published on it was over 2.9 million as of May 2021. Of these, more than 2.5 million apps are classified as regular apps, while 0.4 million apps are classified as low-quality apps by AppBrain. Android's worldwide popularity makes it a more attractive target for cybercriminals and is more at risk from malware and viruses. Studies have proposed various methods of detecting these attacks, and ML is one of the most prominent techniques among them. This is because ML techniques are able to derive a classifier from a (limited) set of training examples. The use of examples thus avoids the need to explicitly define signatures in developing malware detectors. Defining signatures requires expertise and tedious human involvement and for some attack scenarios explicit rules (signatures) do not exist, but examples can be obtained easily. Numerous industrial and academic research has been carried out on ML-based malware detection on Android, which is the focus of this review paper.

Malware Classification is the process of assigning a malware sample to a specific malware family. Malware within a family shares similar properties that can be used to create signatures for detection and classification. Signatures can be categorized as static or dynamic based on how they are extracted. Regression is a statistical method used in finance, investing, and other disciplines that attempts to determine the strength and character of the relationship between one dependent variable and a series of other variables. Neural networks, also known as artificial neural networks (ANNs) or simulated neural networks (SNNs), are a subset of machine learning and are at the heart of deep learning algorithms. Their name and structure are inspired by the human brain, mimicking the way that biological neurons signal to one another. The classification network selects the category based on which output response has the highest output value. Classification neural networks become very powerful when used in a hybrid system with the many types of predictive neural networks.

1.1 Android Architecture

Android is built on top of the Linux Kernel. Linux is chosen because it is open source, verifies the pathway evidence, provides drivers and mechanisms for networking, and manages virtual memory, device power, and security. Android has a layered architecture. The layers are arranged from bottom to top. On top of the Linux Kernal Layer, the Hardware Abstraction Layer, Native C/C++ Libraries and Android Runtime, Java Application Programming Interface (API) Framework, and System Apps are stacked on top of each. Each layer is responsible for a particular task. For example, the Java API Framework provides Java libraries to perform a location awareness application-related activity such as identifying the latitude and the longitude.

Android-based applications and some system services use the Android Runtime (ART). Dalvik was the runtime environment used before the ART. Both ART and Dalvik were created for the Android applications-related projects. The ART executes the Dalvik Executable (DEX) format and the bytecode specification. The other aspects are memory management and power management since the Android-based applications run on battery-powered devices with limited memory. Therefore, the Android operating system is designed in a way that any resource can be well managed. For instance, the Android OS will automatically suspend the application in memory if an application is not in use at the moment. This state is known as the running state of the application life cycle. By doing this, it can preserve the power that can be utilised when the application reopens. Otherwise, the applications are kept idle until they are closed.

1.2 Signs of Malware on Android

Malware often does not make itself obvious. You might first notice signs and symptoms that your device has been infected before taking further action to confirm the presence of malware and finally to get rid of it.

- Your phone tends to overheat. Malware can strain your phone by consuming high amounts of RAM and CPU power. Malware on Android typically runs in the background, consuming more power, and leading to a rise in your phone's temperature. However, an overheating phone doesn't necessarily mean a malware infection. Phones can overheat for various reasons, such as bad software updates or high ambient temperatures.

- The battery drains faster than usual. By constantly running in the background, malware consumes more of your phone's resources—this constant use of power results in faster battery drain.

- Malware attacks are the most common case that can be identified as a threat to Android. There are various definitions for malware given by many researchers depending on the harm they cause. The ultimate meaning of the malware is any of the malicious application with a piece of malicious code which has an evil intent to obtain unauthorised access and to perform neither legal nor ethical activities while violating the three main principles in security: confidentiality, integrity, and availability.

- Malware related to smart devices can be classified into three perspectives as attack goals and behaviour, distribution and infection routes, and privilege acquisition modes. Frauds, spam emails, data theft, and misuse of resources can be mentioned as the attack goals and behaviour perspective. Software markets, browsers, networks, and devices can be identified as the distribution and infection routes. Technical exploitation and user manipulation such as social engineering can be listed under the privilege and acquisition modes. Malware specifically related to the Android operating system is identified as Android malware which harms or steals data from an Android-based mobile device. These are categorised as Trojans, Spyware, adware, ransomware, worms, botnet, and backdoors. Google describes malware as potentially harmful applications. They classified malware as commercial and noncommercial spyware, backdoors, privilege escalation, phishing, types of frauds such as click fraud, toll fraud, Short Message Service (SMS) fraud, and Trojans.

- You've noticed a slowdown in your phone's performance. A slow device could indicate outdated software or a lack of storage space. However, a potential cause is also malware. We've touched on how malware consumes many resources, which has a side effect of a slower phone as less RAM and CPU power are available for the OS and other apps.

- Persistent pop-up ads. The sudden onset of pervasive ads indicates that you've installed an app of unknown or nefarious origin that pushes unwanted advertisements to your device.

- Unrecognized apps. If you see apps on your phone that you don't remember installing, there's a high chance they're viruses.

- Unexplained increase in data usage. If you've found that your data usage has increased sharply on your latest phone bill, chances are there's a malicious code or program sending data from your phone to unknown servers.

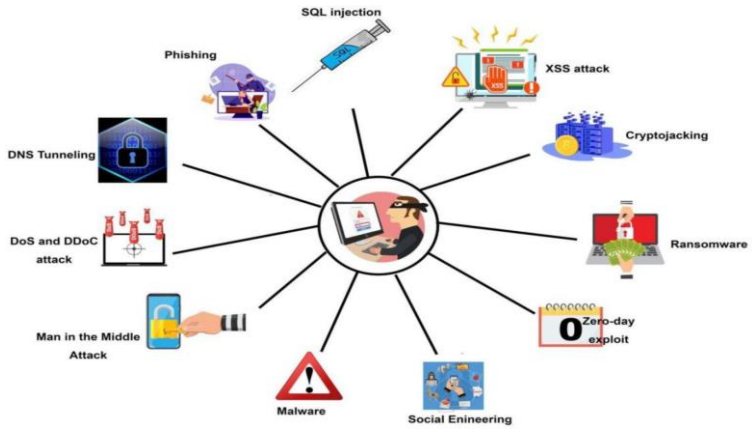

Figure: 1 Signs of Malware

2. Literature Review

Attackers who make malicious applications have come up with new methods of targeting victims of Android users. So, many researchers have verified the effectiveness of the available tools or suggested new tools that may be more effective in the process of accurately detecting malicious applications. Abdulrahman et al. offers a new mechanism for detecting malicious Android applications by using pseudodynamic analysis and constructing an API call graph for each execution path, based on a deep learning model built and trained on a data set consisting of approximately 30 thousand malicious apps and 25 thousand benign apps. However, the researchers used an API call graph to symbolize all viable execution paths that malware may track over its running time. Therefore, transform it into a low dimension numeric vector feature set to be inserted into the deep neural network. They also centered on maximizing the network efficiency by evaluating completely different embedding algorithms and tuning various network configuration parameters to guarantee that the hyper-parameters best assortment has obtained to reach the highest accuracy results. Their results are summarized as that the offered malware classification is reached at 98.86% level in accuracy, 98.65% in F-measure, and 98.47%, 98.84% within the recall and precision, respectively. Suleiman et al. proposed a classification approach based on parallel machine learning to detect Android malware. Depends on real malware samples and benign applications have derived from it, a total of 179 training features were extracted and divided into API calls and commands related: 54 features. App permissions: 125. A composite classification model was developed from a parallel set of heterogeneous classifiers, namely Simple Logistic, Naive Bayes, Decision Tree, PART, and RIDOR. Their results are summarized that PART managed to outperform all the other classifiers, as it achieved true-positive rate 0.95%, true-negative rate 0.96%, false-positive rate 0.03%, falsenegative rate 0.04%, accuracy 0.96%.

3. Methodology

Android was first released in 2008. A few years later, the security concerns were discussed with the increasing popularity of Android applications. More attention was received towards applying ML for software security in the last five years because many researchers continuously identify and propose novel ML-based methods. This review was conducted according to the Preferred Reporting Items for Systematic reviews and Meta-Analysis (PRISMA) model. The database usage and inclusion and exclusion criteria were also defined at this stage. The study selection criteria were defined to identify the studies aiming to answer the formulated research questions as the third stage. We reviewed threats to the validity of the review and the mechanism to reduce the bias and other factors that could have influenced the outcomes of this study as the last step of the review process. Deep Learning (DL) is defined as learning and improving by analysing algorithms on their own.

3.1 How to Detect Malware on Android

Seeing signs of malware on your Android phone? The next step is to get further confirmation that malware has made its way into your device. Here's how to do that. Check for side-loaded apps on your Android phone. You can install apps from outside the Play Store on an Android phone. However, you should avoid downloading apps from apps or browsers as Google cannot screen them, and they are more likely to contain malware.

3.2 How to Remove Malware on Android

Reboot your Android phone in Safe Mode. If you suspect your phone is infected with malware and it isn't working as expected, Safe Mode is worth a shot. Safe Mode, also known as recovery mode, reboots your device to its default state from when you purchased it. Safe Mode doesn't load third-party apps, widgets, or any customizations you might have. It allows you to turn off third-party apps and services, including malware that may have been installed.

3.3 Enable Safe Mode with These Steps: `

1. Press and hold down the power key. Depending on your device, you may have to press the power button and volume up button at the same time.

2. Touch and hold the on-screen power button until the **Reboot to safe mode** message appears.

3. Tap on **OK** to reboot your Android phone in Safe Mode.

3.4 How to Prevent Malware on Your Android Phone

A malware attack can happen to anyone. There are some actions that Android phone users can take to protect themselves. These include:

- **Use the full range of cyber security tools.** Tools like an antivirus program and a VPN for Android are your line of defense against cyber attacks
- **Keep your phone up to date.** Keeping your device software updated is one of the easiest ways to ensure you have a head start in thwarting the latest malware infections and fixing software bugs. Consider enabling automatic updates.
- **Don't open strange attachments or unknown links.** Never trust attachments of unknown origin. In fact, don't trust all attachments sent to you by trusted sources.
- At least until you've verified their authenticity, as phishing scams are extremely common.
- **Only install apps from trusted sources.** By installing apps from the Play Store, you can ensure that they are authenticated and scanned by Google.
- **Periodically review your apps and their permissions.** By reviewing your apps, you can keep aware of your installed apps and the permissions you've granted them. This way, you can spot unrecognized apps and take swift action to uninstall them.
- **Use strong passwords.** A strong password is your first defense against unauthorized access to your accounts. Pair it with two-factor authentication for an additional layer of security. The ultimate password power move is to use a password manager. Password managers generate strong passwords, securely store them, and automatically fill them into login screens.

3.5 Machine Learning

Organizations worldwide are scrambling to incorporate machine learning into their operations, as a result, opportunities for aspiring data scientists are multiplying. One of the most intriguing capabilities of machine learning techniques is the – Ransom ware detection using machine learning algorithms like Regression, Neural networks and classification.

Malware detection in Android can be performed in two ways; signature-based detection methods and behaviour-based detection methods. The signature-based detection method is simple, efficient, and produces low false positives. The binary code of the application is compared with the signatures using a known malware database. However, there is no possibility to detect unknown malware using this method. Therefore, the behaviour-based/anomaly-based detection method is the most commonly used way. This method usually borrows techniques from machine learning and data science. Many research studies have been conducted to detect Android malware using traditional ML-based methods such as Decision Trees (DT) and Support Vector Machines (SVM) and novel DL-based models such as Deep Convolutional Neural Network (Deep-CNN) and Generative adversarial networks. These studies have shown that ML can be effectively utilised for malware detection in Android. Most of these studies used datasets such as Drebin, Google Play, AndroZoo, AppChina, Tencent, YingYongBao, Contagio, Genome/MalGenome, VirusShare, IntelSecurity/MacAfee, MassVet, Android Malware Dataset (AMD), APKPure, Anrdoid Permission Dataset, Andrototal, Wandoujia, Kaggle, CICMaldroid, AZ, and Github to perform experiments and model training in their studies.

Algorithms are taught using statistical techniques to produce classifications or predictions and to find important insights in data mining projects. As the name suggests, it gives modern devices the power to analyze data and learn, which empowers them with the human-like ability to analyze, understand and perform complex tasks.

3.6 Malware Detection Using Machine Learning Techniques

All malware detection methods can be classified as signature-based or behavior-based. It is essential to comprehend the principles of the two malware analysis methodologies, static analysis, and dynamic analysis, before diving into these techniques. Static analysis**,** as the name implies, is performed ‘’statically,’ that is, without running the file while dynamic analysis is done by executing the file on a virtual machine. Static analysis can be thought of as “reading” the malware’s source code to deduce the file’s behavioral properties. Various techniques used in the static analysis:

- **File Format Inspection** Metadata in files can be particularly useful. For instance, PE files may offer a plethora of data regarding build time, functions that are imported and exported, and more.
- String Extraction is the process of scrutinizing software output and extracting information about malware operation.
- Fingerprinting entails performing cryptographic hash computations and locating environmental artifacts such as hardcoded usernames, filenames, and registry strings.
- AV inspection scanners will identify any well-known malware in the examined file. Although it might seem trivial, antivirus and sandboxes commonly employ this kind of detection to “confirm” their findings.
- Disassembly involves translating machine code into assembly language to deduce the logic and intents of the software. The most popular and reliable approach of static analysis is this one.

Dynamic analysis is yet another type of analysis. The behavior of the file is observed during execution, as opposed to static analysis, and the attributes and intents of the file are deduced from that data.

Normally, the file is executed in a virtual setting, like a sandbox. This kind of analysis allows for the discovery of all behavioral characteristics, including opened files, produced mutexes, and other things. It is also comparatively faster than static analysis.

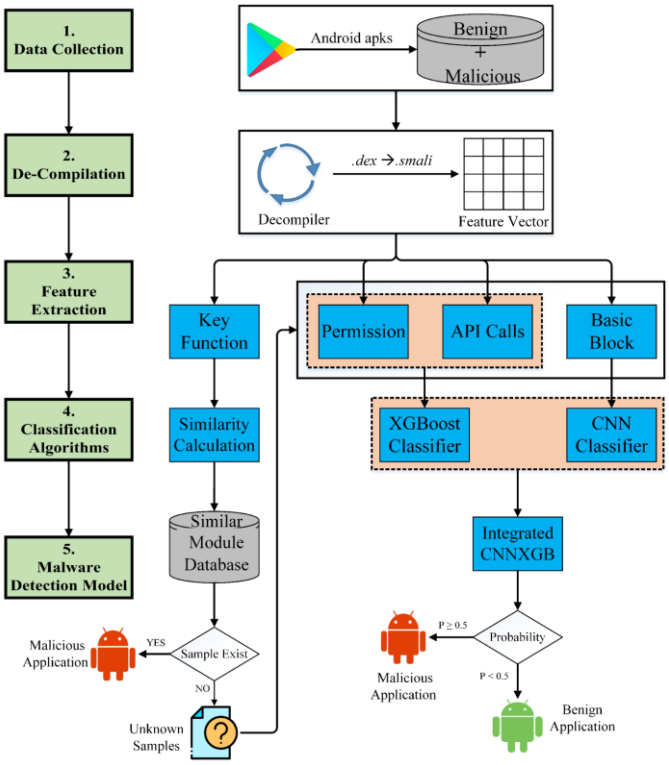

Figure: 2 Malware Detection

3.7 How Machine Learning Works In Malware Detection in Antiviruses

Machine Learning has improved the ability to offer defense against the most recent, never-before-seen malware. These expert systems provide real-time visibility and context into attacks, allowing Windows Defender AV to provide real-time protection against a wide range of threats.

Context-aware detonation systems collect massive amounts of threat intelligence by analyzing millions of potential malware samples. Cloud security engines receive this threat knowledge as input, enabling real-time attack detection and prevention. In addition to Java malware, windows defender also searches for the payloads, which are typically Java remote access Trojans (RATs) such as Jrat and Qrat or online banking Trojans. Organizations must continually monitor and correlate millions of internal and external data elements from their user base and infrastructure to stay safe from cyber-attacks. However, it's easier said than done since it's impossible to monitor and process such a huge amount of information manually. In situations like these, machine learning excels, because it can swiftly identify trends and foresee dangers in various types of data sets.

Cyber teams may identify risks more rapidly and separate circumstances that call for more in-depth human study by automating the analytic process. Einfochips provides all-round cyber security services for threat modelling, VAPT (Vulnerable Assessment Penetration Testing) devices, OS/firmware, web/mobile applications, data, and cloud workloads to detect and classify malware using Machine Learning techniques like Signature-based algorithm, Feature Extraction, Static Analysis and Dynamic Analysis etc .Moreover, many tools like Virustotal, Process monitor, Regshot, Wireshark, Procmon, etc. are used to classify malware types like trojans, backdoors, types of viruses and worms, Rootkits etc. This approach ultimately assists customers in deploying secure products in the open world, which helps protect products from Malicious Software.

Malware detection and classification work best with large datasets that contain all current malware

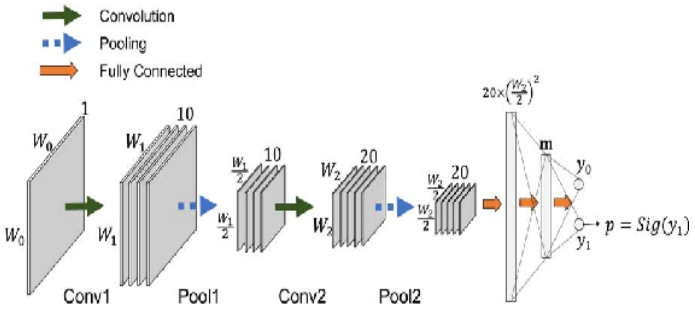

Fig. 4. Structure of the CNN.

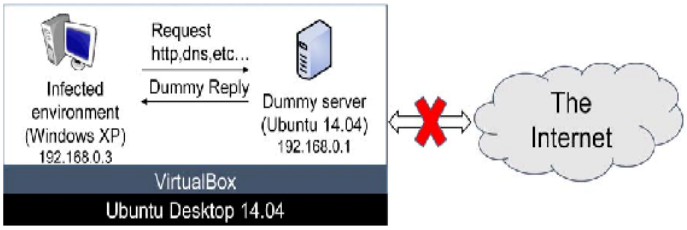

Figure: 3 Machine Learning Works in Malware Detection

Regression is a statistical method used in finance, investing, and other disciplines that attempts to determine the strength and character of the relationship between one dependent variable denoted and a series of other variables. Classification is a supervised machine learning method where the model tries to predict the correct label of a given input data. In classification, the model is fully trained using the training data, and then it is evaluated on test data before being used to perform prediction on new unseen data types.

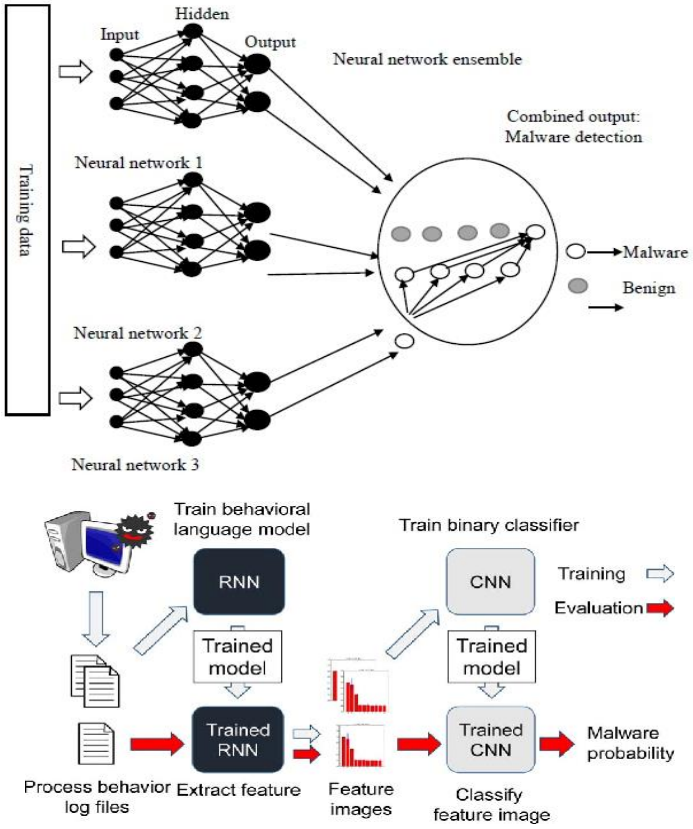

Figure: 4 Malware Detection in Antiviruses

4. Performance Analysis

4.1 Comparisons of impact of the code coverage on machine learning classifiers

In this section the performance of the three input generation approaches are compared using seven popular machine learning classifiers. From the experiments we can gain insight into the impact of their relative code coverage capacities on machine learning-based Android malware detection performance.

First, we discuss the performance evaluation results from Dataset1 (i.e., 1146 malware and 1109 benign samples). All results are obtained using 10 fold cross-validation approach. Figure 9 summarizes the weighted F-measures (W-FM) of the seven classifiers for the three input generation methods. The Random Forest (RF) classifier performs best for the three methods. The state-based method achieved the best W-FM of 0.943, followed by the hybrid method with 0.934 and then the random-based method with 0.926.

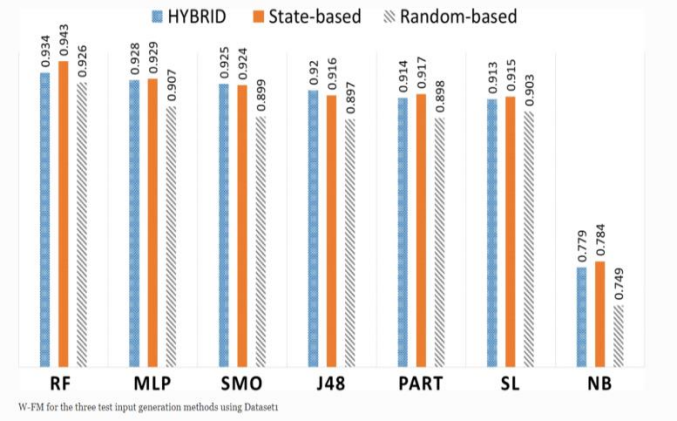

Figure: 5 Comparisons with Machine Learning classifiers

4.2 Comparison of Frameworks Using Manifest and Dex Code Analysis.

This section summarises the works that carried out manife.st analysis only. The summary clearly shows permissions as the main feature extracted from the manifest file. We also observed a relatively low accuracy rate reported. Generally, the authors used imbalanced and relatively smaller datasets while performing manifest file analysis. Furthermore, the review also indicated that Random Forest is the go to ML algorithm when permissions are used

ML Algorithm	Features	Dataset	Accuracy
Random Forest	Permissions	200 and 500 samples	91.75% 91.58%
Naive Bayesian Classification	Permissions	231 benign 246 malware	89%
Simple Logistic, NaiveBayes, BayesNet, SMO, IBK, J48, Random Tree, Random Forest (WEKA)	uses-feature uses-permissions	1811 benign 4301 malware	Random Forest (50) 86.41%
SVM, Random Forest and Decision Trees	Permissions	310,926 benign 4868 malware	Random Forest 94.62%
Logistic Regression	Permissions	Drebin and Gnome Project Datasets	88.28%
SVM and 67 more (WEKA)	Significant Permissions	From 310,926 benign 5494 and 54,694 malware	98.81%

Figure: 6 Comparison of Frameworks using Manifest Analysis.

According to the research on ML-based techniques to malware detection, 65 percent of studies utilised static analysis, 15% used dynamic analysis, and the other 20% employed a mixed analytic strategy. Because of the benefits it delivers, such as the ability to uncover more vulnerabilities, localise issues, and save money, static analysis may be more tempting than dynamic analysis. Table 3 shows the chart representation of ML algorithms commonly used for Android malware detection.

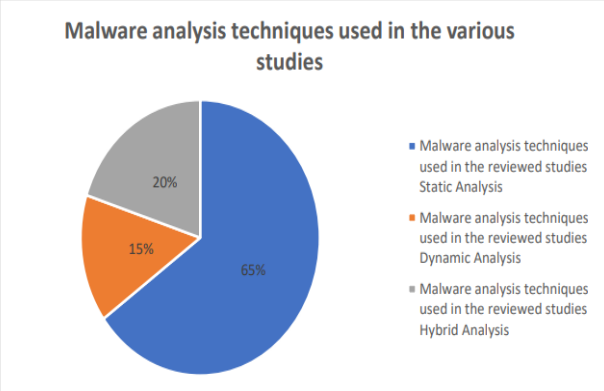

Figure: 6 Comparison of Frameworks using Manifest Analysis.

5. Conclusion

The large number of research works dealing with Android malware detection, which usually report high-performance metrics using a wide variety of ML algorithms can be used to tag the problem as *solved* and demotivate further research. Malware has rapidly become a significant security threat for the computing community, which becomes one of the reasons for most of the current security problems on the Internet. Although a considerable amount of research effort has gone into malware detection, however, malicious code remains a vital threat on the Internet today. Of recent, various Malware detection techniques and approaches

have been proposed to tackle these problems. Unfortunately, these techniques and approaches have some shortcomings that deter them from eliminating the problem. It is vital to design a framework that can correctly identify malware as the number of threats posed to Android devices rises every day. These threats are transmitted mostly through malicious applications or malware. To produce the most optimised feature subset that may be utilised to train machine learning algorithms in the most efficient manner, the proposed technique employs an evolving Genetic Algorithm.

6. References

[1] Pallavi V. Baviskar, Guddi Singh, Vijay Narendranath Patil, "Design of Machine Learning-Based Malware Detection Techniques in Smartphone Environment", International Conference for Advancement in Technology (ICONAT), pp.1-5, 2023.

[2] A. Syed Musthafa K. Sankar T, "A hybrid machine learning technique for early prediction of lung nodules from medical images using a learning-based neural network classifier", concurrency and Computation Practice and Experience, Volume35, Issue3, 2023, 1-15

[3] Mohammed N. AlJarrah, Qussai M. Yaseen, Ahmad M. Mustafa, "A Context-Aware Android Malware Detection Approach Using Machine Learning", *Information*, vol.13, no.12, pp.563, 2022.

[4] Anish, T.P, Syed Musthafa, A., Elavarasu, R., Block chain Based Secure Data Transmission among Internet of Vehicles, "International Conference on Innovative Practices in Technology and Management", ICIPTM 2022, 2022, pp. 765–769

[5] Senthil Kumar, R., Syed Musthafa, A, "Recursive CNN Model to Detect Anomaly Detection in X-Ray Security Image", International Conference on Innovative Practices in Technology and Management, ICIPTM 2022, 2022, pp. 742–747

[6] S. Zeadally, F. Siddiqui and Z. Baig, "25 years of bluetooth technology", *Futur. Internet*, 2019.

[7] Syed Musthafa, A, Dhananjayan, D, "Smart Authentication System Using Deep Learning Techniques Based on Face and License Plate Recognition", International Conference on Advanced Computing and Communication Systems, ICACCS 2022, 2022, pp. 1240–1244

[8] R. Mayrhofer, J. Vander Stoep, C. Brubaker and N. Kralevich, "The Android Platform Security Model", *ACM Trans. Priv. Secur*, 2021.

[9] Jerome Nithin Gladson, G.,Syed Musthafa A, Gopinathan, R, "Study on the performance of a flat plate solar water heater using a hybrid nanofluid", Materials Today: Proceedings, 2022, 69, pp. 1145–1149

[10] S. Shakya, "An Efficient Security Framework For Data Migration In A Cloud Computing Environment", *J. Artif. Intell. Capsul. Networks*, 2019.

[11] Suresh, R., Syed Musthafa, A, Vishwakarma, S, "Analyzing thermal performance of a solar PV using a nanofluid", Materials Today: Proceedings, 2022, 69, pp. 1126–1129

[12] M. S. R, "Soft Computing Based Autonomous Low Rate Ddos Attack Detection and Security for Cloud Computing", *J. Soft Comput. Paradig*, 2019.

[13] Syed Musthafa A, Monisha, B "Aadhar UID Mask Detecting Tool Using CNN With Verhoeff Algorithm", International Conference on Advanced Computing and Communication Systems, ICACCS 2023, 2023, pp. 792–796

[14] D. S. A, "Enhanced Soft Computing Approaches For Intrusion Detection Schemes In Social Media Networks", *J. Soft Comput. Paradig*, 2019.

[15] Rajasekar P, Syed Musthafa, A, "Perceptual Video Summarization Using Key frames Extraction Technique", 3rd International Conference on Innovative Practices in Technology and Management, ICIPTM 2023, 2023

[16] B. Vivekanandam, "Design an Adaptive Hybrid Approach for Genetic Algorithm to Detect Effective Malware Detection in Android Division", *J. Ubiquitous Comput. Commun. Technol*, 2021.

[17] Syed Musthafa, A. ,Sankar, "A hybrid machine learning technique for early prediction of lung nodules from medical images using a learning-based neural network classifier", Concurrency and Computation: Practice and Experience This link is disabled., 2023, 35(3), e7488

[18] Srivastava, S., Munjal, N, Syed Musthafa A, "Unravelling the gait and balance: A novel approach for detecting depression in young healthy individuals", Journal of Intelligent and Fuzzy Systems This link is disabled., 2023, 45(6), pp. 12079–12093

[19] P. Saravanan, J. Selvaprabu, L. Arun Raj, A. Abdul Azeez Khan and K. Javubar Sathick, "Survey on Crime Analysis and Prediction Using Data Mining and Machine Learning Techniques", *Advances in Smart Grid Technology*, pp. 435-448, 2021.

[20] Syed Musthafa Akbar Batcha, Dilip Kumar Sharma, Mohanraj Elangovan,, "Securing data in transit using data-in-transit defender architecture for cloud communication", Soft Computing (2021), ISSN: 14327643, 14337479, doi.org/10.1007/s00500-021-05928-6, june 2021

[21] R Jane Preetha Princy, Saravanan Parthasarathy, P Subha Hency Jose, Arun Raj Lakshminarayanan and Selvaprabu Jeganathan, "Prediction of Cardiac Disease using Supervised Machine Learning Algorithms", *2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS)*, 2020.

[22] Syed Musthafa, A., Ravi, Logesh, Palani, Saravanan, "A Fuzzy based High-Resolution Multi-View Deep CNN for Breast Cancer Diagnosis through SVM Classifier on Visual Analysis", Journal of Intelligent & Fuzzy Systems, IOS Press, 10.3233/JIFS-189174, Page 1-14, September 2020

[23] K. Chumachenko, *Machine Learning Methods for Malware Detection and Classification*, 2017.

[24] Syed Musthafa, Roy Setiawan, Dac-Nhuong Le, "Utilizing Index-Based Periodic High Utility Mining to Study Frequent Itemsets", Arabian Journal for Science and Engineering, Page 1-9, Springer Berlin Heidelberg, 2021

[25] L. Liu, B. S. Wang, B. Yu and Q. X. Zhong, "Automatic malware classification and new malware detection using machine learning", *Frontiers of Information Technology & Electronic Engineering*, vol. 18, no. 9, pp. 1336-1347, 2017.

[26] Syed Musthafa A, "Securing data in transit using data-in-transit defender architecture for cloudcommunication", Soft Computing, springer, 6, 2021

[27] Masabo, K. S. Kaawaase and J. Sansa-Otim, "Big data: deep learning for detecting malware", *Proceedings of the 2018 International Conference on Software Engineering in Africa*, pp. 20-26, May 2018.

[28] A. Syed Musthafa, "An Optimized Fuzzy based Ant Colony Algorithm for 5G-MANET", Computers, Materials & Continua, Vol 70, Issue 1, PP 1069-1087, Tech Science Press, 2021

[29] Garg, S.; Peddoju, S.K.; Sarje, A.K. Network-based detection of Android malicious apps. Int. J.Inf. Secur. 2017, 16, 385–400. [CrossRef]

A, Syed Musthafa, "Experimental Methodology to Optimize Power Flow in Utility Grid with Integrated Renewable Energy and Storage Devices Using Hidden Markov Model", Electric Power Components and Systems, 2023

[30] Khan, J.; Shahzad, S. Android Architecture and Related Security Risks. Asian J. Technol. Manag.

[31] Res. [ISSN: 2249–0892] 2015, 5, 14–18. Available online: http://www.ajtmr.com/papers/Vol5Issue2/Vol5Iss2_P4.pdf (accessed on 19 May 2021).

[32] Ramesh, Musthafa, A. Syed," Packet disordering prediction based on neural networking by integrated sliding", American Institute of Physics Conference Series, November 2023, DOI:10.1063/5.0180565