Detection of Malware in Android Application using Machine Learning

Paka Sudarsan, V.N V S K Vinay, Rejeti Pavan Satya Kumar, Burugupalli Jagadeesh,
Guide : Maddipati Satya Srinivas, Associate Professor
Department of Computer Science and Engineering
Sasi Institute of Technology and Engineering

Abstract **- Malware is a piece of software which contains malicious data which damage or disrupt a device's normal use it is intentionally created to exploit systems without the user's knowledge. Since there is a rapid increase in mobile usage especially android ones has changed the way user access information and perform daily tasks as there is a large increase in human usage, malware also gets proportionately increased With the increasing complexity and diversity of malware, it is difficult for traditional methods to identify them This study explain nature of malware and various forms, including viruses, worms, Trojans, ransomware, adware, spyware, and rootkits. Where each attack has a different way of injecting malware into android environment. Since traditional methods tries to identify known malware signatures, they tend to fail in predicting new attacks The primary objective of this paper is to utilise ML models especially logistic regression as it is a binary classification model which can handle classification problems well, and make a effective malware prediction model.**

Index Terms - Malware, machine learning, android applications, logistic regression

INTRODUCTION

Malware is a virus which is send by intruder where the primary purpose is to cause harm or exploit systems without the user's knowledge. Thereby stealing sensitive information or gaining unauthorized access to devices and networks. There are various types of malwares, and they are described below:

Viruses: It is defined as a malicious software that spreads from one system from another where it get attached to files, and gets executed while host application is running these viruses gets spread from one system to another, typically through file sharing there are various symptoms which helps in identifying virus affected or not they are the system's speed gets slower , applications run in slow manner often suffers with network problems, unwanted popup windows gets shown on computer screen as they are sign of malware. Programs which are executing without intervention then there is high chance of virus affected system, AS there are viruses where specific applications are targeted which results on log out of their accounts, systems mail often gets attacked by huge number of mails, suddenly system gets crashed down, also have high chance of modifications in home page these viruses gets triggered only when user intervention is made.

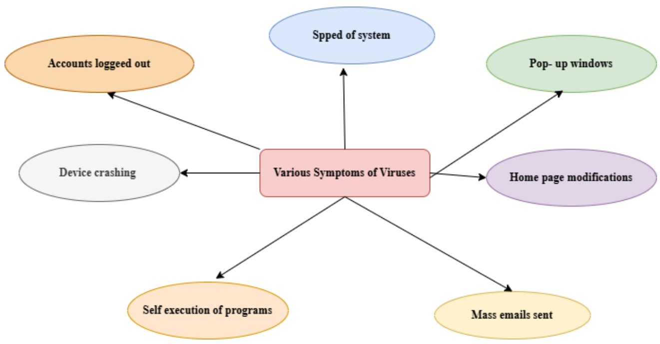

Fig.1. Various symptoms of viruses

As a result of this symptoms various virus occurs and they are described below:

Resident virus: In this type of viruses host computer gets infected by malicious virus by infecting applications.

Multipartite virus: This virus uses multiple methods to infect the system s application and remain in the computer's memory to infect the hard disk this result in system lag these types of viruses can be avoided by not opening attachments from untrusted sources, links from unknown mails, file attachments.

Direct action: In this virus main memory of system gets infected which results in lagged performance of system as the files gets deleted or altered since these viruses have capability to destroy all data on hard disks. Such virus can be avoided by using antivirus software.

Browser hijacker: In these cases, web browsers settings get altered where there will be a modification in homepage replacement, default search engine gets changed since it cannot affect any files it is not treated as virus however it can highly damage computer.

Overwrite virus: They were considered as highly dangerous as it allows intruder to inject malicious code into web-pages which allows intruder to attack high traffic websites they also cause damage to server files.

Worms: It Is malware which gets spread automatically over a network unlike traditional virus where user interaction is required to spread. These worms cause significant damage to network infrastructure by consuming bandwidth and overloading servers. They are associated with payload which delete files, steal data, or install backdoors for future attacks.

Trojans: it is a malicious program because they behave as legitimate software to be perceived by users, once they get installed then they can perform harmful actions, such as stealing data or granting unauthorized access to the infected device. They are considered dangerous as they create a backdoor which allows them for future attacks. Unlike viruses or worms, Trojans do not self-replicate. They rely on users to download and execute them.

Ransomware: It is virus in which users' system gets attacked and damaged through a malware that encrypts the victim's data and to restore access demand a payment usually in cryptocurrency, to restore access. The malware usually locks the victim's files or system by encrypting them. This is highly dangerous as ransomware groups not only demand payment for decryption but also threaten to leak sensitive information unless a second ransom is paid.

Spyware: It is special type of malware unlike rest of them where malware occurred can be known whereas in this It operates in the background, often without the user's knowledge, collecting sensitive data such as login credentials, browsing habits, or financial information. The information gathered is through a keystroke mechanism.

Adware: This is not considered as serious malware where the primary purpose is to display unwanted advertisements although they are not malicious, they degrade the user experience and, in some cases, lead to more dangerous malware infections. This can be done by tracking user behavior, such as browsing history, to display targeted advertisements.

Botnets: Often referred as zombies that are controlled by a central attacker, known as the "botmaster. Where ethe primary purpose is to create large number of server requests which are distributed to multiple systems causing denial of service attacks

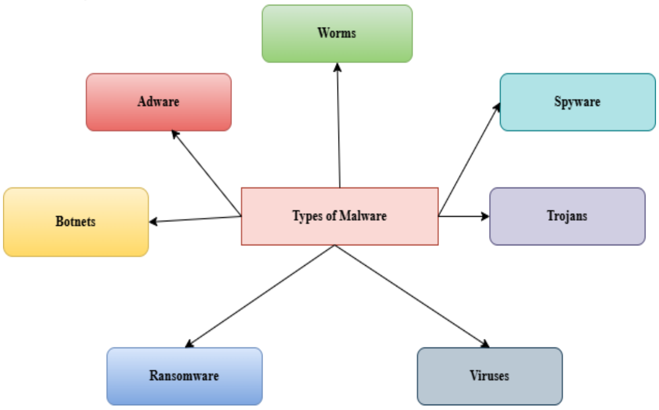

Fig.2. Types of Malware Attacks

LITERATURE SURVEY

Zarni et al. [1] utilised permission-based approach to detect malware as part of this a comprehensive analysis is made where various types of malwares, such as viruses, worms, trojans, spyware, and adware, each having different behaviours and targets. It is common to give permission access to android applications, but it will be misused to access sensitive data or perform certain actions. and the model gets trained by analysing various permissions requested by applications. by identifying them as benign or malicious. and obtained an accuracy of 925 due to feature extraction capability.

Yerima et al. [2] identified increasing number of malware attacks and proposed a Bayesian classification which a statistical method which uses past knowledge and make predictions. initially after data collection feature extraction is made through list of permissions requested by the application , filters through which giving insight into the app's functionality. is checked and obtained an accuracy of 90%.

Arshad et al. [3] made a comprehensive survey on various malware attacks which targets android environment and appropriate measures since android is highly adoptive made it highly vulnerable to various types of malware where in order to prevent this static analysis is made in which it involves examining the code or binary of an Android application without executing it. The focus is on analysing the app's structure, code patterns, permissions, and other attributes. on the on the other side dynamic analysis observing the behaviour of an application during runtime. to monitor any malicious activity. like network traffic, Monitoring CPU usage. By combining both of these make an hybrid approach which offers higher detection accuracy and robustness compared to using static or dynamic techniques alone. And obtained an accuracy of 95%.

McLaughlin et al. [4] utilised dl to detect malware attacks using CNNN initially data is considered form Android marketplaces, including official (e.g., Google Play Store) and third-party sources known to host malware. then statistical analysis is made to extract relevant features from APK. as Static analysis does not require running the app but rather examines the application's code, configuration files, and permissions. then identified the permission requested by app and detects a controlled flow mechanism graph in which any suspicious activity can be identified then by using these data CNN model adapts them as they have hierarchical representation which helped to obtain an accuracy of 92%.

Omer et al. [5] provided a comprehensive analysis of all the existing methods in malware detection where it is identified that there are certain challenges while implanting malware detection mechanism in which Malware developers use techniques like obfuscation, encryption, and dynamic code loading to evade detection by hiding malicious code within legitimate-looking code., due to data imbalance where there are fewer malicious samples compared to benign samples, due to which class imbalance occurs and it is identified that hybrid approach which combines both statistical and dynamic analysis has shown good performance and obtained an accuracy of 96%.

Yuan et al. [6] utilised dl and proposed a DNN to identify by making a behavioural patterns of Android apps. where the primary focus is on permissions requested by the apps and API calls used during execution as they are important in identifying the model observe pattern and overserve these to distinguish between normal and abnormal behaviours. and obtained an accuracy of 97%.

Tong et al. [7] utilised both static and dynamic analysis techniques and proposed a hybrid approach where it captures characteristics (static) and actual runtime behaviour (dynamic), increasing the accuracy and robustness of malware detection. and obtained an accuracy of 95%.

Damshenas et al. [8] utilised ML and identified a detection model focuses on behavioural analysis, which involves monitoring the interaction of apps with the system resources such as network traffic, file system, memory usage, and processor activity., then the ML model extracts feature from dynamic behaviour logs during app execution and trains a classifier to differentiate between benign and malicious applications. and obtained an accuracy of 92%.

Zhu et al. [9] utilised dl and focused on mining sensitive data usage patterns, such as accessing location, contacts, messages, etc. then CNN model is fed with these sensitive data using a sequence of API calls DeepFlow analyses these API sequences to identify patterns that are commonly associated with malicious apps and tried to identified malicious class by obtaining an accuracy of 95%.

Kakavand et al. [10] utilised ML and made a comparative analysis where the features such as permissions, API calls, and system logs. Are taken then created a comprehensive dataset combining static features (e.g., permissions, APK structure) and dynamic features (e.g., system calls, network activity). then model gets trained on this final evaluation is made and RF obtained high accuracy of 93% due to its ensemble approach.

METHODOLOGY

A. Dataset details

Android Malware Dataset for Machine Learning: This dataset is useful for analysing and experimentation in the field of Android malware detection. It contains various features extracted from Android applications, enabling the application of machine learning techniques to classify applications as benign or malicious. This dataset contains 215 attributes extracted from 15,036 applications (5,560 malware apps from Drebin project and 9,476 benign apps). Features are categorised into Manifest permissions, API Calls, Intents.

B. Proposed Method

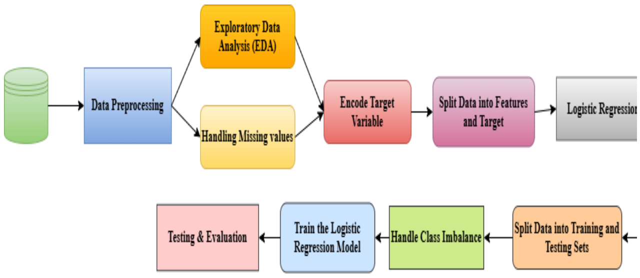

Fig.3. Architecture of the proposed Model

Below is the step-by-step implementation of the above model:
Step 1: Initially after importing necessary libraries dataset gets loaded
Step 2: To the data preprocessing must be done so missing values are identified and dropped then Eda is done to ensure visualizing the data to better understand its structure, trends, and patterns.
Step 3: Then target variables are encoded by Mapping the target variable 'class' (benign or malicious) to numerical values (0 for benign and 1 for malware).
Step 4: Separate the dataset into features (X) and the target variable (y).
Step 5: Defined a logistic regression model and Split Data into Training and Testing Sets
Step 6: Using Random Over Sampling to balance the training dataset by increasing the number of samples in the minority class (malware).
Step 7: Instantiate the logistic regression model, train it on the training data, and measure the training time.
Step Testing and evaluating on unseen data by generating classification report to access performance of the model
The below Fig.4. illustrated how different features are corelated using a heatmap

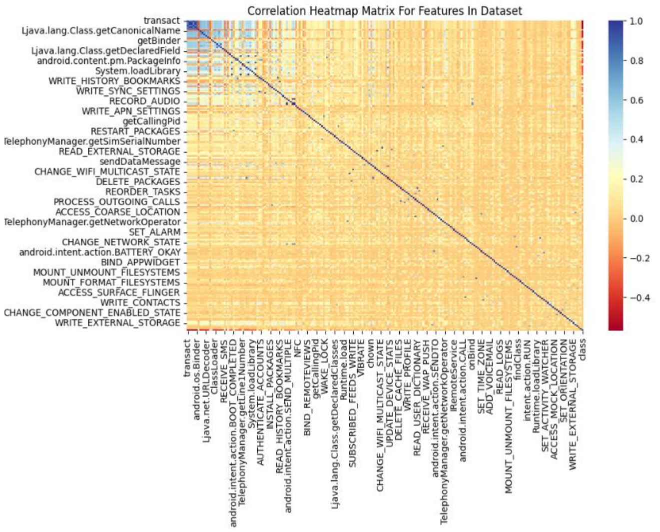

Fig.4. Heatmap Illustration

C. Comparative Analysis

TABLE 1. A OVERVIEW OF COMPARTIVE ANALYSIS OF VARIOUS MODELS

Year	Author Name	Proposed Work	Proposed Algorithm	Accuracy Obtained
2020	Qiu et al.	A survey of Android malware detection with deep neural models.	Deep Neural Models	95%
2018	Xu et al.	DeepRefiner: Multi-layer Android malware detection system applying	Deep Neural Networks	94%

		deep neural networks.		
2017	Hou et al.	Automatic Android malware detection using deep neural networks.	Deep Neural Networks	96%
2020	Lu et al.	Android malware detection based on a hybrid deep learning model.	Hybrid Deep Learning Model	92%
2019	Wang et al.	Effective Android malware detection with a hybrid model based on deep autoencoder and CNN.	Deep Autoencoder + CNN	91%
2019	Naway & Li	Using deep neural networks for Android malware detection.	Deep Neural Network	94%
2019	Masum & Shahriar	Droid-NNet: Deep learning neural network for Android malware detection.	Deep Learning Neural Network	93%
2018	Li et al.	DeepDetector: Android malware detection using deep neural network.	Deep Neural Network	97%
2019	Lee et al.	SeqDroid: Obfuscated Android malware detection using stacked convolutional and recurrent networks.	Stacked CNN + RNN	98%
2022	Naeem et al.	A deep convolutional neural network stacked ensemble for malware threat classification in IoT.	Deep CNN Stacked Ensemble	96%

EXPERIMENTAL SETUP

This experiment requires python version of 3.8 with all the necessary libraries like NumPy, pandas, matplotlib, Grayscale or binary images of different resolutions and complexities were used for testing. This will ensure a good environment makes this suitable for Malware detection in android.

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 04 | April - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

RESULTS DISCUSSION

Logistic regression have performed in detecting malware this is due to statistical approach by learning relationship between the input features and the log-odds of the probability of the positive class, which can capture complex relationships if the features themselves are well-chosen or transformed. Additionally, it is les prone to overfitting since it contains few parameters and handled nonlinearity by using exploratory data analysis which improved data quality and obtained an accuracy of 97%.

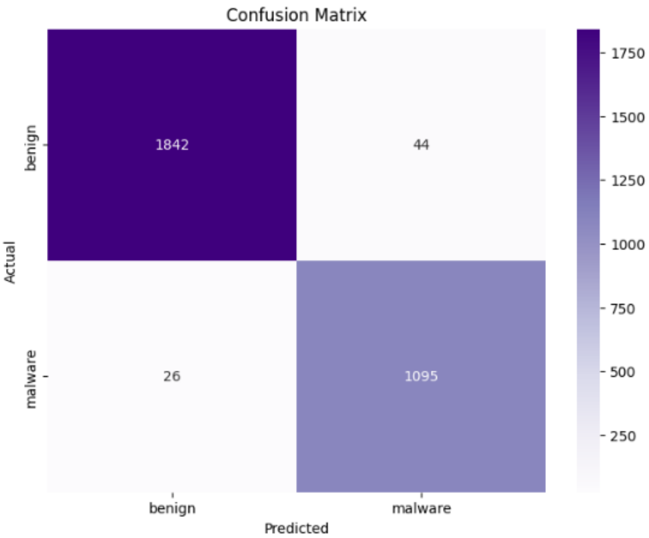

Fig.5. Confusion Matrix Obtained

CONCLUSION & FUTURE SCOPE

Malware detection is a key aspect especially in android applications as there usage is getting increase attack also getting increase on a same level this proposed study utilised ML based logistic regress ion which utilised statistical approach and enhanced raw data by feature extraction eliminating the need for extensive manual feature engineering, Also handled imbalanced datasets where benign samples are more than malicious samples by using oversampling and synthetic data have been successfully employed to mitigate this issue, ensuring that the models are adequately trained to recognize rare malware instances.in the future utilising advanced deep neural networks is necessary to handle even more complicated data in order to make decisions at a faster pace.

.

REFERENCES

[1] Zarni Aung, W. Z. (2013). Permission-based android malware detection. International Journal of Scientific & Technology Research, 2(3), 228-234.

[2] Yerima, S. Y., Sezer, S., McWilliams, G., & Muttik, I. (2013, March). A new android malware detection approach using bayesian classification. In 2013 IEEE 27th international conference on advanced information networking and applications (AINA) (pp. 121-128). IEEE.

[3] Arshad, S., Shah, M. A., Khan, A., & Ahmed, M. (2016). Android malware detection & protection: a survey. International Journal of Advanced Computer Science and Applications, 7(2).

[4] McLaughlin, N., Martinez del Rincon, J., Kang, B., Yerima, S., Miller, P., Sezer, S., ... & Joon Ahn, G. (2017, March). Deep android malware detection. In Proceedings of the seventh ACM on conference on data and application security and privacy (pp. 301-308).

[5] Omer, M. A., Zeebaree, S. R., Sadeeq, M. A., Salim, B. W., Rashid, Z. N., & Haji, L. M. (2021). Efficiency of malware detection in android system: A survey. Asian Journal of Research in Computer Science, 7(4), 59-69.

[6] Yuan, Z., Lu, Y., Wang, Z., & Xue, Y. (2014, August). Droid-sec: deep learning in android malware detection. In Proceedings of the 2014 ACM conference on SIGCOMM (pp. 371-372).

[7] Tong, F., & Yan, Z. (2017). A hybrid approach of mobile malware detection in Android. Journal of Parallel and Distributed computing, 103, 22-31.

[8] Damshenas, M., Dehghantanha, A., Choo, K. K. R., & Mahmud, R. (2015). M0droid: An android behavioral-based malware detection model. Journal of Information Privacy and Security, 11(3), 141-157.

[9] Zhu, D., Jin, H., Yang, Y., Wu, D., & Chen, W. (2017, July). DeepFlow: Deep learning-based malware detection by mining Android application for abnormal usage of sensitive data. In 2017 IEEE symposium on computers and communications (ISCC) (pp. 438-443). IEEE.

[10] Kakavand, M., Dabbagh, M., & Dehghantanha, A. (2018, November). Application of machine learning algorithms for android malware detection. In Proceedings of the 2018 International Conference on Computational Intelligence and Intelligent Systems (pp. 32-36)

[11] Qiu, J., Zhang, J., Luo, W., Pan, L., Nepal, S., & Xiang, Y. (2020). A survey of android malware detection with deep neural models. ACM Computing Surveys (CSUR), 53(6), 1-36.

[12] Xu, K., Li, Y., Deng, R. H., & Chen, K. (2018, April). Deeprefiner: Multi-layer android malware detection system applying deep neural networks. In 2018 IEEE European

Symposium on Security and Privacy (EuroS&P) (pp. 473-487). IEEE.

[13] Hou, S., Saas, A., Chen, L., Ye, Y., & Bourlai, T. (2017, July). Deep neural networks for automatic android malware detection. In Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2017 (pp. 803-810).

[14] Lu, T., Du, Y., Ouyang, L., Chen, Q., & Wang, X. (2020). Android malware detection based on a hybrid deep learning model. Security and Communication Networks, 2020(1), 8863617.

[15] Wang, W., Zhao, M., & Wang, J. (2019). Effective android malware detection with a hybrid model based on deep autoencoder and convolutional neural network. Journal of Ambient Intelligence and Humanized Computing, 10, 3035-3043.

[16] Naway, A., & Li, Y. (2019). Using deep neural network for Android malware detection. arXiv preprint arXiv:1904.00736.

[17] Masum, M., & Shahriar, H. (2019, December). Droid-NNet: Deep learning neural network for android malware detection. In 2019 IEEE International Conference on Big Data (Big Data) (pp. 5789-5793). IEEE.

[18] Li, D., Wang, Z., & Xue, Y. (2018, June). Deepdetector: android malware detection using deep neural network. In 2018 International Conference on Advances in Computing and Communication Engineering (ICACCE) (pp. 184-188). IEEE.

[19] Lee, W. Y., Saxe, J., & Harang, R. (2019). SeqDroid: Obfuscated Android malware detection using stacked convolutional and recurrent neural networks. Deep learning applications for cyber security, 197-210.

[20] Naeem, H., Cheng, X., Ullah, F., Jabbar, S., & Dong, S. (2022). A deep convolutional neural network stacked ensemble for malware threat classification in internet of things. Journal of Circuits, Systems and Computers, 31(17), 2250302.

[21] Gao, C., Cai, M., Yin, S., Huang, G., Li, H., Yuan, W., & Luo, X. (2023). Obfuscation-resilient android malware analysis based on complementary features. IEEE Transactions on Information Forensics and Security.

[22] Millar, S., McLaughlin, N., Martinez del Rincon, J., Miller, P., & Zhao, Z. (2020, March). Dandroid: A multi-view discriminative adversarial network for obfuscated android malware detection. In Proceedings of the tenth ACM conference on data and application security and privacy (pp. 353-364).

[23] Kumar, S., & Panda, K. (2023). SDIF-CNN: Stacking deep image features using fine-tuned convolution neural network models for real-world malware detection and classification. Applied Soft Computing, 146, 110676.

[24] Zou, K., Luo, X., Liu, P., Wang, W., & Wang, H. (2020). ByteDroid: android malware detection using deep learning on bytecode sequences. In Trusted Computing and Information Security: 13th Chinese Conference, CTCIS 2019, Shanghai, China, October 24–27, 2019, Revised Selected Papers 13 (pp. 159-176). Springer Singapore.

[25] Qiu, J., Zhang, J., Luo, W., Pan, L., Nepal, S., & Xiang, Y. (2020). A survey of android malware detection with deep neural models. ACM Computing Surveys (CSUR), 53(6), 1-36.

[26] Roy, K. S., Ahmed, T., Udas, P. B., Karim, M. E., & Majumdar, S. (2023). MalHyStack: a hybrid stacked ensemble learning framework with feature engineering schemes for obfuscated malware analysis. Intelligent Systems with Applications, 20, 200283.

[27] Zhang, N., Xue, J., Ma, Y., Zhang, R., Liang, T., & Tan, Y. A. (2021). Hybrid sequence-based Android malware detection using natural language processing. International Journal of Intelligent Systems, 36(10), 5770-5784.

[28] Surendran, R., Thomas, T., & Emmanuel, S. (2020). A TAN based hybrid model for android malware detection. Journal of Information Security and Applications, 54, 102483.

[29] Wang, W., Ren, C., Song, H., Zhang, S., & Liu, P. (2022). Fgl_droid: an efficient android malware detection method based on hybrid analysis. Security and Communication Networks, 2022(1), 8398591.

[30] Zhu, H. J., Wang, L. M., Zhong, S., Li, Y., & Sheng, V. S. (2021). A hybrid deep network framework for android malware detection. IEEE Transactions on Knowledge and Data Engineering, 34(12), 5558-5570.