Ilkogretim Online - Elementary Education Online, 2021; Vol 20 (Issue 1): pp. 7963-7970
http://ilkogretim-online.org
doi: 10.17051/ilkonline.2021.01.821

Amdetect : Android Malware Detection Using Machine Learning

SUMALATHA POTTETI Assistant Professor, Department of Computer Science and Engineering, Bhoj Reddy Engineering College for Women, Hyderabad,Telangana, India.

Dr. G. S. MAHALAKSHMI Associate Professor, Department of Computer Science and Engineering, Anna University, Chennai, Tamilnadu, India.

Abstract: The basic idea behind malware is to take advantage of a victim's computer resources. Recent malware evolution has made it more resilient and adaptable to accomplish a variety of objectives, including anonymity for illicit activity, sensitive data theft, and denial of service (DoS). But generally, economics is the driving force. Malware families have created a broad range of methods to get money, from straightforward blackmail via a DoS threat to sophisticated bank trojans, with the hope of eventually making some fiduciary money. Cybercriminals look for new models in this unstoppable evolution in order to make rapid money. This method works really well with digital money. In recent years, protecting Android mobile and systems against cyberattacks has become increasingly important. Even though the majority of systems today are constructed with enhanced security features, there are still a significant number of vulnerabilities, mostly brought about by old software, unsecured protocols and systems, and human mistake. malware detection in android mobiles can take on many different forms and aim for any infrastructure, including cloud computing, Mobile and Internet of Things (IoT) devices.

Keywords: Android, machine learning, decision tree, random forest.

1. Introduction:

Threats to a computer network and its assets originate from malware attackers' attempts to breach defences and protective layers. Since they can offer some level of security on computer networks and systems to identify and mitigate malware attacks, anti-malware softwares have been employed extensively in businesses for a long time. However, many anti-malware programmes often use hashing algorithms, network connection whitelisting, or static string matching technique. These solutions are too basic to deal with sophisticated malware attacks, which might incorporate evasive strategies to conceal communication channels and evade the majority of detection methods. The problem has created a significant risk to an organization's security and is a huge challenge that needs to be solved.

Malware attackers try to breach defensive measures and layers of protection, posing risks to a computer network and its assets. Since they can offer some level of security on computer networks and systems to identify and mitigate malware attacks, anti-malware softwares have been employed extensively in businesses for a long time. However, a lot of anti-malware programmes often use network communication whitelisting, hashing algorithms, or static string matching techniques. These solutions are too basic to deal with sophisticated malware attacks, which might incorporate evasive strategies to conceal communication channels and evade the majority of detection methods. The problem has created a significant risk to an organization's security and is a huge challenge that needs to be solved.

Malware attackers' attempts to get past defenses and protective layers are the source of threats to a computer network and its assets. Anti-malware software's have been used widely in businesses for a long time since they can provide some level of security on computer networks and systems to identify and neutralize malware attacks. However, many anti-malware programmers frequently employ static string matching, network connection whitelisting, or hashing methods. These systems are too simplistic to handle sophisticated malware attacks, which may use evasive tactics to mask communication channels and avoid most detection techniques. The issue is a serious difficulty that has to be resolved and has significantly increased the risk to an organization's security.

2. Related Work:

From the perspective of feature extraction, the current research on Android malware detection can be separated into static analysis[1] and dynamic analysis[2]. Analysis of the source code or of the features that have been taken out of the source code is referred to as static analysis. Without running the programme, this technique can examine the source code [3][4]. Decompilation, reverse analysis, pattern matching, and static system call analysis are all included in the static analysis. Static analysis has the benefits of low resource usage, quick detection, and less real-time processing, but the drawback is that the detection accuracy isn't very high.

In modern research, the static analysis method is the one that is most frequently employed. In [5] design of security rules employs a signature-based strategy to identify the application under scrutiny. MADAM is a host-based malware detection solution for Android devices that [6] suggested. MADAM examines and correlates features simultaneously at the kernel level, application level, user level, and package level in order to identify and stop malicious activity In [7] framework usage proposals included strings, API calls, permissions, and other features to reflect the various application characteristics from various angles. Even though malware has many characteristics that are similar to those of innocuous programmes, their existence-based and similarity-based feature vector generation methods are quite good at telling malware and benign applications apart. The abstracted API calls of function methods are also kept by [8] who then calculate the confidence of association rules between the abstracted API calls to create a set of abstracted API calls transactions. Create a detection system by combining machine learning to recognise the various behavioural patterns. To identify malware from the

viewpoint of behaviour, the MaMaDroid framework [9] builds the sequence seen in the API call graph as a Markov chain.
A series of techniques based on examining an application's runtime behaviour is referred to as dynamic analysis [10]. To track an application's use of the network, system calls, files, memory, information access patterns, and processing characteristics, it is typically essential to execute it in a certain environment [11]. By examining whether the aforementioned characteristics are typical, dynamic analysis determines whether an application is harmful. The benefit of dynamic analysis [12] is that it can examine an application based on its malware-like behaviour and is unaffected by code obfuscation and encryption. However, it uses up more system resources and necessitates analysts [13] with advanced technical skills, making it unsuitable for use in testing on a big scale.

3. Machine Learning models

Artificial intelligence, machine learning, and deep learning are the three main scientific fields in information exploration and hidden data set trends identification. Applications for machine learning that integrate statistical mining with result optimization include bioinformatics, biometrics, machine vision, computer anatomy and forensics, facial identification, the detection of fraud as well as falsification, and the economy of handwriting, to name just a few. Machine learning focuses on training clever models to replicate accurately and fast.

We looked at many supervised machine and deep learning classifiers in an effort to identify the models that perform the best by utilizing the unique qualities of each one. We choose to employ deep learning architectures based on Fully Connected Neural Networks since we do not presume either a temporal or spatial relationship among the features to be input to models (FCNNs). We also took into account a number of other well-known machine learning methods, including Random Forests, CART, and C4.5, as well as Logistic Regression models, and Classification And Regression decision Trees (CART and C4.5).

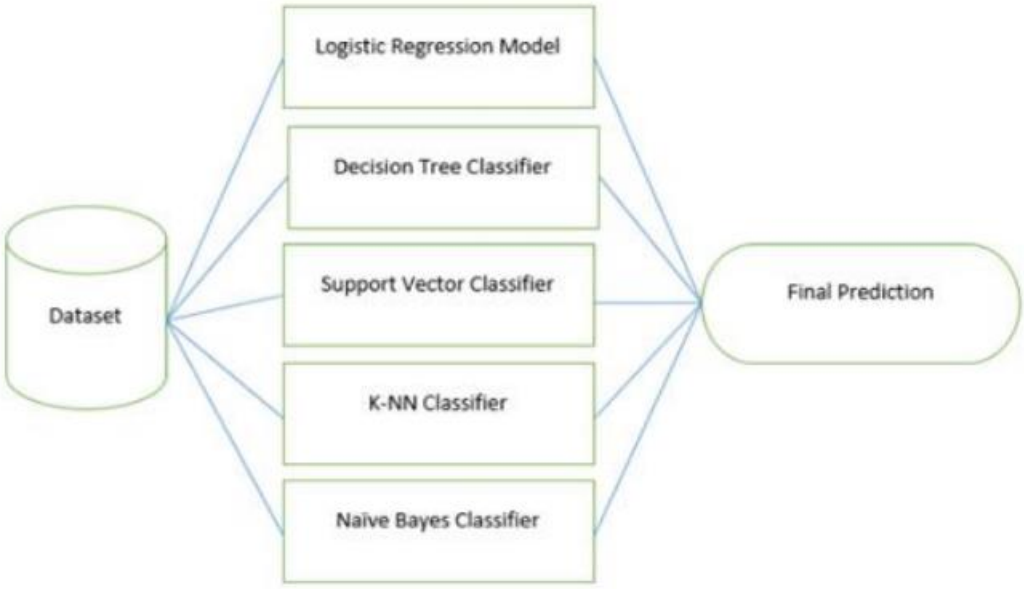

Figure.1 Machine Learning Approaches

The Logistic Regression model, which permits modeling of binary outcomes with only one of two potential values, is essentially a classification technique. Several works suggest using this technique to classify Internet traffic based on protocols.
Using a tree of rules learned from both discrete and continuous variables, the Classification Regression Tree (CART) and the C4.5 algorithm are decision tree models capable of classifying data into distinct groups. These models enable the classification of an input feature vector by indicating the likelihood that it belongs to a specific class based on the likelihood of the tree's eventual leaf.

4. Analysis and Discussion:

Machine learning makes predictions and learns from data. For instance, consumer expectations are based on prior interactions with similar clients. The machine is more hungry for trustworthy and performance-conscious learning than it is for its own learning. Deep learning also goes by the moniker of deep formal learning. Researchers can build their own function extractions for machine learning by using a variety of mathematical methods and theoretical formulations.

In order to achieve a more accurate classification using a majority vote procedure through bootstrap aggregating (bagging) and random selection features for each tree, the Random Forest (RF) model combines many Decision Trees (particularly, CART trees). Due to the RF Model's non-linear classification capabilities, effectiveness, and robustness, it is widely employed.

Algorithm	Scaling Factor	Tuning Needed	Features Detection	Effectiveness of Small Data
KNN	Yes	Minimal	No	No

Linear regression	No (unless regularized)	None (excluding regularization)	No	Yes
Logistic regression	No (unless regularized)	None (excluding regularization)	No	Yes
Naive Bayes	No	Some for feature extraction	Yes	Yes
Decision trees	No	Some	No	No
Random Forests	No	Some	Yes	No

Table.1 Analysis of Machine Learning Algorithms for malware detection

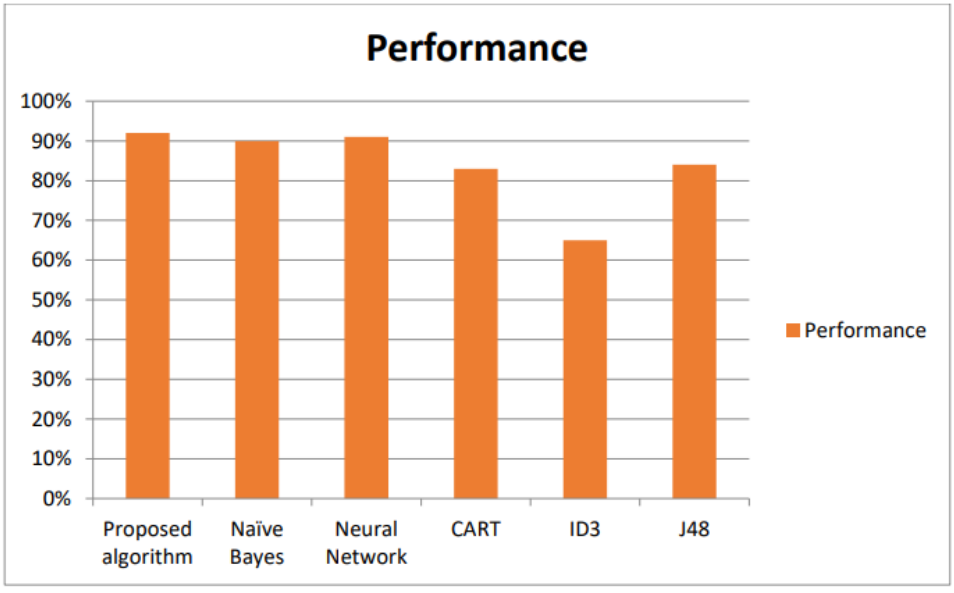

Figure.2 Performance Analysis of ML Approaches

5. Conclusion:

The number of apps that can be categorized as malware is always growing, and as new malware variants and camouflage strategies are being developed all the time, it is crucial for users and the third-party application markets to successfully detect malware in a short amount of time. There are still issues to be resolved regarding how to increase detection accuracy and decrease detection time. We presented machine learning based malware detection approaches, a system for quickly detecting Android malware that combines machine learning features with permission features from various operation levels to create feature vectors. The machine learning technique is used for feature selection to lower the method's feature dimensionality and time complexity. Machine learning approaches and classifier that has been suggested in this work, is used in this study to carry out malware detection and family classification.

References:

[1] Arp, D., Spreitzenbarth, M., Hubner, M., Gascon, H., Rieck, K., & Siemens, C. E. R. T. (2014, February). Drebin: Effective and explainable detection of android malware in your pocket. In Ndss (Vol. 14, pp. 23-26).

[2] Amdani and Ajani S., S.Y., 2020. Dynamic Path Planning Approaches based on Artificial Intelligence and Machine Learning. Turkish Journal of Computer and Mathematics Education (TURCOMAT), 11(3), pp.2084-2098.

[3] Batista, F. K., Martin del Rey, A., & Queiruga-Dios, A. (2020). A New Individual-Based Model to Simulate Malware Propagation in Wireless Sensor Networks. Mathematics, 8(3), 410.

[4] Buczak AL, Guven E. A survey of data mining and machine learning methods for cyber security intrusion detection. IEEE Communications Surveys & Tutorials. 2016 Oct;18(2):1153-76.

[5] Chen, S., Xue, M., Fan, L., Ma, L., Liu, Y., & Xu, L. (2019, February). How Can We Craft Large-Scale Android Malware? An Automated Poisoning Attack. In 2019 IEEE 1st International Workshop on Artificial Intelligence for Mobile (AI4Mobile) (pp. 21-24). IEEE.

[6] Christodorescu, S. Jha, S. A Seshia, E Bryant, “Semantics-aware malware detection”, IEEE Symposium on Security and Privacy, pp. 32-46, May 2005. [20] Christodorescu, S. Jha, S. A Seshia, E Bryant, “Semantics-aware malware detection”, IEEE Symposium on Security and Privacy, pp. 32-46, May 2005.

[7] Landage, Prof. M. P. Wankhade, “Malware and Malware Detection Techniques: A Survey”, International Journal of Engineering Research & Technology (IJERT), Vol. 2 , Issue 12, pp. 1-8 , December 2013.

[8] Landage, Prof. M. P. Wankhade, “Malware and Malware Detection Techniques: A Survey”, International Journal of Engineering Research & Technology (IJERT), Vol. 2 , Issue 12, pp. 1-8 , December 2013.

[9] Li, J., Sun, L., Yan, Q., Li, Z., Srisa-an, W., & Ye, H. (2018). Significant permission identification for machine-learning-based android malware detection. IEEE Transactions on Industrial Informatics, 14(7), 3216-3225.

[10] 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). ACM.

[11] S. N. Ajani, "Probabilistic path planning using current obstacle position in static environment," 2nd International Conference on Data, Engineering and Applications (IDEA), 2020, pp. 1-6, doi: 10.1109/IDEA49133.2020.9170727.

[12] Milosevic N, Dehghantanha A, Choo KK. Machine learning aided android malware classification. Computers & Electrical Engineering. 2017 Jul 1;61:266-74.

[13] Milosevic, N., Dehghantanha, A., & Choo, K. K. R. (2017). Machine learning aided Android malware classification. Computers & Electrical Engineering, 61, 266-274.

[14] S. Alam, R. N. Horspool and I. Traore,"MARD: A framework for metamorphic malware analysis and real-time detection", 28th IEEE International Conference on Advanced Information Networking and Applications (AINA), Victoria, pp. 480- 489, May 2014.

[15] S. Das, Y. Liu, W. Zhang, and M. Chandramohan ,"Semantics-Based Online Malware Detection: Towards Efficient Real-Time Protection Against Malware", IEEE transactions on information forensics and security, Vol. 11, Issue 2, pp. 289 - 302 , February 2016.

[16] V. M. Afonso, D. S. F. Filho, A. R. A. Gregio, P. L. de Geus, M. Jino, "A hybrid framework to analyze web and operating system malware", IEEE International Conference on Communications (ICC), Ottawa, pp. 966-970, June 2012.

[17] V. M. Afonso, D. S. F. Filho, A. R. A. Gregio, P. L. de Geus, M. Jino, "A hybrid framework to analyze web and operating system malware", IEEE International Conference on Communications (ICC), Ottawa, pp. 966-970, June 2012.

[18] M. Wanjari, Ajani S. "An Efficient Approach for Clustering Uncertain Data Mining Based on Hash Indexing and Voronoi Clustering," 2013 5th International Conference and Computational Intelligence and Communication Networks, 2013, pp. 486-490, doi: 10.1109/CICN.2013.106.

[19] Vinayakumar, R., Soman, K. P., Poornachandran, P., & Sachin Kumar, S. (2018). Detecting Android malware using long short-term memory (LSTM). Journal of Intelligent & Fuzzy Systems, 34(3), 1277-1288.

[20] Wang, S., Chen, Z., Yan, Q., Yang, B., Peng, L., & Jia, Z. (2019). A mobile malware

detection method using behavior features in network traffic. Journal of Network and Computer Applications, 133, 15-25.

[21] Wang, W., Li, Y., Wang, X., Liu, J., & Zhang, X. (2018). Detecting Android malicious apps and categorizing benign apps with ensemble of classifiers. Future Generation Computer Systems, 78, 987-994.

[22] 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(8), 3035-3043.

[23] Z. Bazrafshan, H. Hashemi, S. M. H. Fard and A. Hamzeh, "A survey on heuristic malware detection techniques", 5th IEEE Conference on Information and Knowledge Technology (IKT), pp. 113-120, May 2013.

[24] Z. Chen, M. Roussopoulos, Z. Liang, Y. Zhang , Z. Chen and A. Deli, "Malware characteristics and Threats on the Internet Ecosystem", Journal of Systems and Software, Vol. 85, Issue 7 , pp.1650-1672, July 2012.

[25] Z. Chen, M. Roussopoulos, Z. Liang, Y. Zhang , Z. Chen and A. Delis, "Malware characteristics and Threats on the Internet Ecosystem", Journal of Systems and Software, Vol. 85, Issue 7 , pp. 1650-1672, July 2012.