ISSN: 2278-4632
Vol-13, Issue-08, August 2023

ADVANCED ANDROID APPLICATIONS DEVELOPMENTS USING DEEP LEARNING ALGORITHMS

Dr. Prasuna Grandhi
Associate Professor
Department of CSE
St. Ann's College of Engineering & Technology, Chirala

Dr. Ratna Raju Mukiri
Associate Professor
Department of CSE
St. Ann's College of Engineering & Technology, Chirala
mukiriratnaraju001@gmail.com

Dr. Hari Kishan Chapala
Professor & HOD
Department of CSE - AIML
St. Ann's College of Engineering & Technology, Chirala

ABSTRACT: The Android operating system has been the most popular for smartphones and tablets since 2012. This popularity has led to a rapid raise of Android malware in recent years. To achieve a secured smart environment for future sustainable computing, android-based smart devices must provide more resilient and attack-resistant commitments, so that the android malware detector, once trained on a dataset, can continue to identify new malware without retraining. Classic signature-based detection techniques fall short when they come up against a large number of users and applications. Machine learning, on the other hand, appears to work well. a code behaviour signature-based malware detection framework mistreatment associate degree SVM rule is planned, which might sight malicious code and their variants effectively in runtime and extend malware characteristics information dynamically. The framework deploys DDQN algorithm to obtain a subset of features is effective malware classification to valid subset of features over a larger range the exploration-exploitation policy is applied in the model training phase. The recurrent neural network (RNN) is used as the decision network of DDQN to give the framework the ability to sequentially select process. Our experiments are based on real-world Apps, and use five different classification algorithms to detect the malware. We propose multi-level of procedures to find most significant permission instead of extracting total permissions. The propose system used supervised classification method in machine-learning to classify number of families of benign and malware apps.

INDEX TERMS: Support Vector Machine, Svm Classifier, Deep learning, Reinforcement Learning Android Malware Detection Feature Selection RNN, Static analysis Dynamic analysis Feature Selection.

ISSN: 2278-4632
Vol-13, Issue-08, August 2023

1. INTRODUCTION

Android operating system is provided by Google is predicted to continue have a dramatic increase in the market with around 1.8 billion Android-based devices to be shipped by 2023 sta. It is currently leading the mobile OS market with over 90% market share compared to Windows, Blackberry, and Symbian OS [1]. The advancement in Data and Communication Technology (DCT) remains a major security model and challenge to achieve sustainability for future computing by preventing and mitigating various known and unknown cyber-attacks [2].These small variations are introduced by replacing a few lines of code instructions keywords. This limitation prevents the anti-malware from detecting variations on back malware and malware based on zero-day attacks [3]. Machine learning algorithms is used in new variety of applications such as in medicine email filtering speech recognition and computer vision where it is difficult or unfeasible to develop conventional algorithms to perform the needed tasks [4].Some researchers utilize state-of-the-art machine learning models like deep learning and online learning or ensemble learning to find multi-class attacks effectively in the Android environment [5]. We propose a hybrid analysis method for the detection of the Android malware that integrates the advantages of static and dynamic analysis methods [6]. Proposed system it is used the advanced engineering tool such as apktool, dex2jar and jdgui for static malware analysis [7]. We propose multilevel data pruning approach to extract most significant permission only [8]. Android malware is developed 3-levels of pruning by mining the permission data to identify the very significant permissions is effective in distinguishing between benign and malicious apps [9].

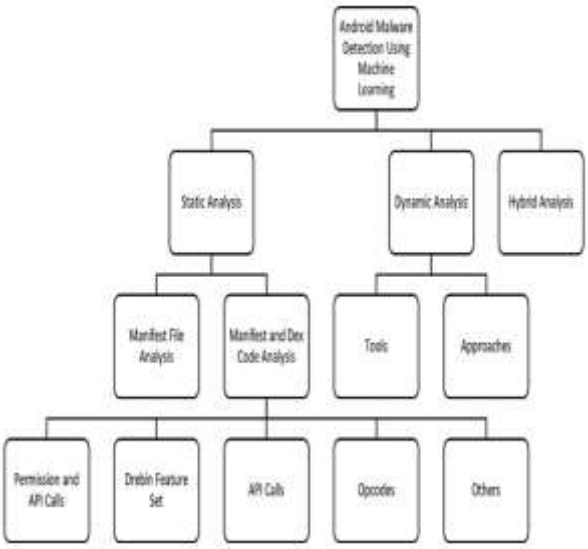

Fig1. Android malware detection techniques svm machine learning.

2. RELATED WORK

A recent survey revealed that smartphone users have substantially increased from 7.643 billion in 2021 to 8.378 billion in 2023 People use these devices for several purposes like making calls and transactions exchanging data and social communications[10] Dynamic analysis approach on the other hand consist of running Android applications in a controlled environment such as an Android Virtual Device (AVD) emulator emu in a real device

ISSN: 2278-4632
Vol-13, Issue-08, August 2023

in order to monitor the apps behaviour [11]. Android is based on a modified version of the Linux Kernel and is mainly used in mobile devices Android runs applications in their dedicated processes [12]. Each application is given a dedicated data directory, which can only be read or written to by the application and no other application has the permission to access it [13]. The discipline of machine learning employs various methods to teach computers tasks where no fully satisfactory algorithm is available. In cases where large numbers of potential answers exist new approach is to label some of the correct answers as valid [14]. Implemented automated defender agent that detected and mitigated on-going attacks based on pre-defined probabilities of success [15]. The proposed works and some gaps such as detecting of unknown malware and reducing of false positive alarm are still remained. This paper proposed multi-level data pruning method in feature selection instead of extracting all permission features [16]. The analysis found that nearly 100,000 apps request no permissions at all protected APIs placed between Applications and Libraries. This permission is defined in Manifest file AndroidManifest.xml, which is compulsory for shipping each android app [17].

3. SYSTEM MODEL

The SVM-based active learning framework for smart phone malware detection and within the mechanical man system valid the effectiveness of the strategy tests show that the planned methodology has sensible relevancy and measurability will be complete on a range of well-liked malware observation and might find unknown malware [18]. The core part of the DroidRL framework is built up by the DDQN-based decision networkthe autonomous agent independently carries out an action decided by the decision network to select one feature into its observed state from the environment using their prior knowledge [19]. In the detection phase the unknown samples are detected by the obtained classifier model is obtained by means of training the hybrid feature vectors [20].Android apps run in separate processes under distinct user identifiers (UIDs) every distinctpermission. Programs can't either read or write each other's data or code of apps and applications must be done explicitly for sharing data [21].

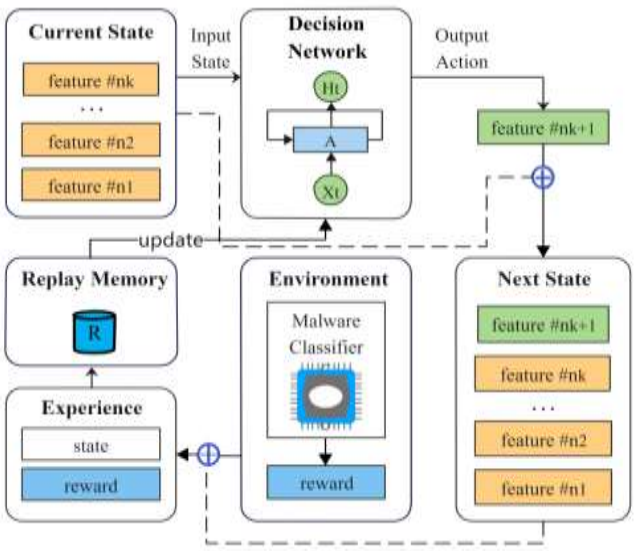

Fig2: DroidRL for feature selection in Android malware detection

4. PROPOSED WORK

ISSN: 2278-4632
Vol-13, Issue-08, August 2023

The proposed work is optimized ensemble learning for android malware detection (OEL-AMD). The proposed system aims to characterize many permissions intents and API call logs generated during installation or before and after device restart by any android application [22]. The Android operating system uses a permission-based model not only to limit the behaviour of an application but also to inform the user of the application potential behaviour.Android makes use of the security features present in the Linux kernel in a typical Linux system user resources are isolated from different users of the system. This results in one user not having access to resources of another user and maintaining a level of security [23].

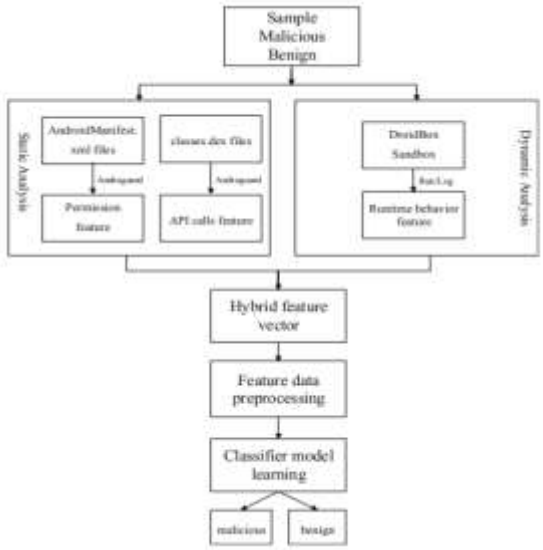

Fig3. Architecture of the proposed detection method

A. DEEP LEARNING CLASSIFIER

Our main goal is to build a model for DL-Droid to enable security classification and find Android malware from stating apps. We train our deep learning classifiers on a classification problem with two methods benign or malicious. We utilize currently supports only the Multilayer Perceptron classifier (MLP). A confusion matrix is performed in our system to evaluate the effectiveness of different classifiers [24]. The second phase of the experiments compared the performance between the proposed DL and seven popular machine learning models isproposed. The classifiers include Support Vector Machine (SVM Linear), Support Vector Machine with radial basis function kernel (SVM RBF), Naive Bayes (NB), Simple Logistic (SL), Partial Decision Trees (PART), Random Forest (RF), and Decision Tree [25]. We also find the performance of each classifier for two different test input generation methods. The results of our experiments using the performance metrics defined.

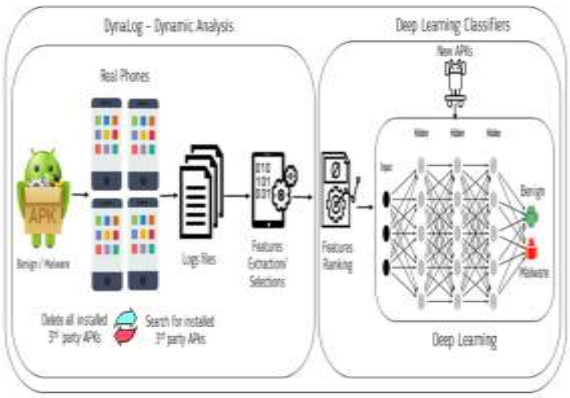

Fig4. DL-Droid framework

B. ALGORITHM: Optimal Feature Selection

For supervised algorithms to work efficiently, users must input the correct data so

ISSN: 2278-4632
Vol-13, Issue-08, August 2023

that the algorithms can learn and make inferences [26]. The most of the intents and permissions recorded from different android malware attacks exhibit a common range for both benign and malignant attacks. The selection of this feature selection algorithm is inspired by its elegant performance in different domains excellent leadership and intelligently hunting of the grey wolves. The hunting task is completed by chasing, encircling, harassing, and attacking [27].

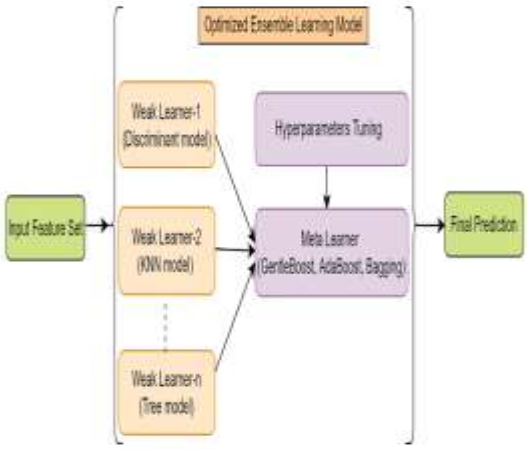

Fig5. Block diagram of optimized security method

Step1. True Positive (TP): The application is malicious and is correctly predicted as such.

Step2. False Positive (FP): The application is benign and is falsely predicted as malicious.

Step3. True Negative (TN): The application is benign and is correctly predicted as such.

Step4. False Negative (FN): The application is malicious and is falsely predicted as benign

C. FEATURES ORDERING

There is a special consideration for applying natural language processing methods to DroidRL feature ordering. Natural language is sequential in nature which means replacing two words in a sentence can make the sentence confusing and meaningless [27].Exchanges positions of any two features in the input state should not influence the result this character of feature selection differs from that of natural language. If this particular property is not addressed it may have a negative impact on RNN-like decision network learning [28].Features represented by one-hot vectors go through the embedding layer and become more dense vectors. These vectors then are fed into the RNN-like network and finally enter a fully connected layer and a softmax layer.

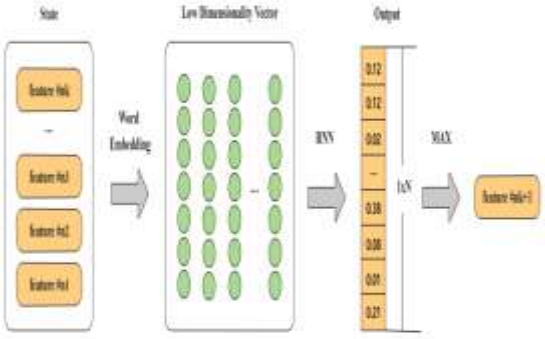

Fig6: DroidRL's Decision Network.

5. EXPERIMENT SETUP

This section provides information on the hardware environment for training DroidRL, our dataset, and hyperparameters settingTraining Environment We performed all the experiments on the server with single Tesla V100 GPU and a CPU with two cores. The GPU was used to accelerate the training of decision network in DroidRL, while the training and prediction of the classifiers in DroidRL used CPU only. After the training processDroidRL's classifier is extracted

ISSN: 2278-4632
Vol-13, Issue-08, August 2023

separately for testing any hardware that capable of running machine learning algorithms. The Benign samples used in this paper are mainly downloaded from the Google Play store to ensure the availability of the experimental data. In this experiment, we randomly selected 150 malicious apps and 150 benign apps.The lowest accuracy when apply algorithm was achieved by NaivesNayes. Furthermore, in an observation based in this work, the finding show that Random Forest is the best parameter at each node in decision tree is made from a randomly selected numbers in feature selectionAfter each training episode, the DroidRL framework was tested by running five testing evaluation episodes . In addition, taking 50 training episodes as a period, the DroidRL framework was tested after every period. After using the average accuracy of 50 episodes, the training classification accuracy more stable in the first 20 periods compared with clearly shows its changing trend.

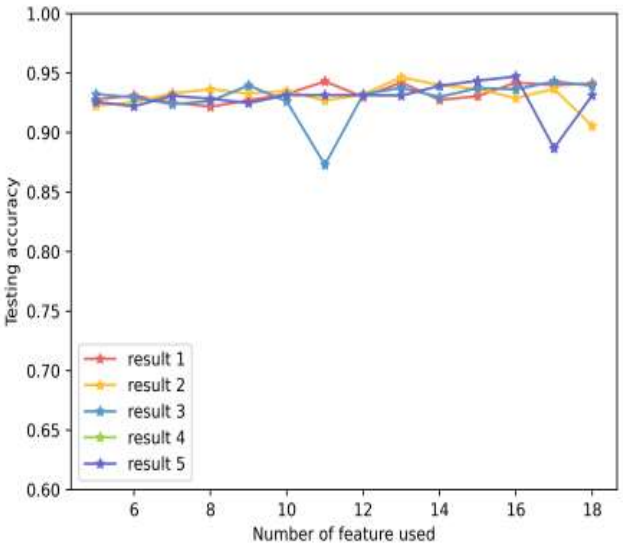

Fig7: Stability verification of different feature selection results by DroidRL

6. CONCLUSIONSAND FUTURE WORK

We presented DL-Droid, an automated dynamic analysis framework for Android malware detection. DL-Droid employs deep learning with a state-based input generation approach. Rapid growth in the use of Android devices is highlighted problem Android malware motivating the development of ML-based malware detection algorithm. ML-based method is principle least identify malware that has not been seen before and therefore have the potential to prevent zero day attacks.The proposed framework extracts permissions from Android applications and further combines the API calls to characterize each application as a high dimension feature vector.Especially, the RNN-like network is applied as the decision network in DDQN for its capability of processing variable-length sequences. For the purpose of finding the correlation between features, DroidRL uses word embedding to semantically represent the features. In future we consider the method of semantics learning into feature extraction to analyse the behaviour of malware. We can further mine the association rules between features select better feature selection algorithms to reduce the redundancy of features and further improve the efficiency of classification.

7. REFERENCES

ISSN: 2278-4632
Vol-13, Issue-08, August 2023

[1] Android (GOOG) Just Hit a Record 88% Market Share of All Smartphones—Quartz. Accessed: Jan. 28, 2022.[Online]. Available: https://qz.com/826672/android-goog-just-hit-a-record-88-market-shareof-all-smartphones/

[2] Kumar P, Gupta GP, Tripathi R. An ensemble learning and fog-cloud architecturedriven cyber-attack detection framework for IoMT networks.ComputCommun 2021 Jan;15(166):110–24.

[3] Stats, S. G. Mobile Operating System Market Share Worldwide. https://gs. statcounter.com/os-market-share/mobile/worldwide (Accessed: Oct. 19, 2021)

[4] Imtiaz SI, urRehman S, Javed AR, Jalil Z, Liu X, Alnumay WS. DeepAMD: Detection and identification of Android malware using high-efficient Deep Artificial Neural Network. Future Generation computer systems. 2021 Feb 1;115: 844-56.

[5] Sihag V, Vardhan M, Singh P. A survey of android application and malware hardening.ComputSci Rev 2021 Feb;1(39):100365.

[6] Sharma K, Gupta BB. Towards privacy risk analysis in android applications using machine learning approaches.Int J E-Services Mobile Appl (IJESMA) 2019 Apr 1; 11(2):1–21.

[7] Wang C, Xu Q, Lin X, Liu S. Research on data mining of permissions mode for Android malware detection. Cluster Comput 2019 Nov;22(6):13337–50.

[8] Cam NT, Pham VH, Nguyen T. Detecting sensitive data leakage via interapplications on Android using a hybrid analysis technique. Cluster Comput 2019 Jan;22(1):1055–64.

[9] Taheri L, Kadir AF, Lashkari AH. Extensible android malware detection and family classification using network-flows and API-calls. In2019 International Carnahan Conference on Security Technology (ICCST) 2019 Oct 1 (pp. 1-8). .

[10] Shafiq M, Tian Z, Bashir AK, Du X, Guizani M. IoT malicious traffic identification using wrapper-based feature selection mechanisms. Comput Security 2020 Jul;1 (94):101863

[11] William Enck, MachigarOngtang, and Patrick McDaniel.2009.On light weight mobile phone application certification.In Proceedings of the 16th ACM conference on Computer and communications security CCS'09).Association for Computing Machinery, New York, NY, USA, 235–245.

[12] Sanz, Borja& Santos, Igor &Laorden, Carlos &UgartePedrero, Xabier&Bringas, Pablo.

[13] B.H. Robbins, (2010), "Non Parametric Tests", B.H. Robbins Scholars Series, Dept. of Biostatistics, Vanderbilt University.

[14] Bostanci, Betul, &ErkanBostanci, (2013), "An evaluation of classification algorithms using McNemars test", Proceedings of Seventh International Conference on Bio-

ISSN: 2278-4632
Vol-13, Issue-08, August 2023

Inspired Computing: Theories and Applications (BIC-TA 2012). Springer, India.

[15] Dietterich, Thomas G, (1998), "Approximate statistical tests for comparing supervised classification learning algorithms." Neural computation 10.7, pp: 1895-1923.

[16] La Polla, Mariantonietta, Fabio Martinelli, & Daniele Sgandurra, (2013), "A survey on security for mobile devices", IEEE communications surveys & tutorials 15.1, pp: 446-471.

[17] Tam, Kimberly, (2017), The evolution of android malware and android analysis techniques", ACM Computing Surveys (CSUR) 49.4, pp: 76

[18] Liang, Shuang, &Xiaojiang Du, (2014), "Permission-combination-based scheme for android mobile malware detection", Communications (ICC), 2014 IEEE International Conference. 53

[19] Saracino, Andrea, (2016), "Madam: Effective and efficient behavior-based android malware detection and prevention", IEEE Transactions on Dependable and Secure Computing.

[20] Linn, Cullen, &SaumyaDebray, (2003), "Obfuscation of executable code to improve resistance to static disassembly", Proceedings of the 10th ACM conference on Computer and communications security, ACM.

[21] Mahindru, A., Sangal, A.L., 2020. SOMDROID: android malware detection by artificial neural network trained using unsupervised learning. Springer Berlin Heidelberg. URL: https://doi.org/10.1007/s12065-020-00518-1, doi:10.1007/s12065-020-00518-1.

[22] Mahindru, A., Sangal, A.L., 2021. FSDroid:- A feature selection technique to detect malware from Android using Machine Learning Techniques: FSDroid. Multimedia Tools and Applications doi:10.1007/s11042-020-10367-w.

[23] Mantoo, B.A., 2020. A hybrid approach with intrinsic feature-based android malware detection using lda and machine learning, in: The International Conference on Recent Innovations in Computing, Springer. pp. 295–306.

[24] Mcwilliams, G., Sezer, S., Yerima, S.Y., 2014. Analysis of bayesian classification-based approaches for android malware detection. Information Security Iet 8, 25–36.

[25] Melo, F.S., 2001. Convergence of Q-learning: A simple proof. Institute Of Systems and Robotics, Tech. Rep , 1–4arXiv:arXiv:1011.1669v3.

[26] Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A.A., Veness, J., Bellemare, M.G., Graves, A., Riedmiller, M., Fidjeland, A.K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., Hassabis, D., 2015. Human-level control through deep reinforcement learning. Nature 518, 529–533. URL: http://dx.doi.org/10.1038/nature14236, doi:10.1038/ nature14236.

ISSN: 2278-4632
Vol-13, Issue-08, August 2023

[27] Molina-Coronado, B., Mori, U., Mendiburu, A., Miguel-Alonso, J., 2023. Towards a fair comparison and realistic evaluation framework of android malware detectors based on static analysis and machine learning.Computers & Security 124, 102996. URL: https://www.sciencedirect.com/science/article/pii/S0167404822003881, doi:https://doi.org/10.1016/j.cose.2022.102996.

[28] Narayanan, A., Chandramohan, M., Chen, L., Liu, Y., 2017. Context-aware, adaptive, and scalable android malware detection through online learning.IEEE Transactions on Emerging Topics in Computational Intelligence 1, 157–175.