

# A COMPARATIVE STUDY ON MACHINE LEARNING AND DEEP LEARNING METHODS FOR MALWARE DETECTION

ESLAVATH RAVI<sup>1</sup>, MUMMADI UPENDRA KUMAR<sup>2</sup>

<sup>1</sup>Research Scholar, Osmania University, Department of CSE, Hyderabad, India

<sup>2</sup>Professor, Muffakham Jah College of Engineering and Technology, Department of CSE, Hyderabad, India

E-mail: <sup>1</sup>eslavathravi@gmail.com, <sup>2</sup>upendra.kumar@mjccollege.ac.in

## ABSTRACT

The advent of Artificial Intelligence (AI) and data science with Machine Learning (ML) and deep learning techniques has paved way for solving many real world problems. Malware detection is one such problem that is solved with AI based solutions. Malicious code that comes through genuine software components or through storage media and networks is termed as malware. This paper has made a reviews of literature for ascertaining the current academic thinking and the methods being used or methods possible to detect malware automatically. The study in this paper is divided into three categories known as ML methods for malware detection, deep learning methods for malware detection and optimization methods for improving malware prediction performance. Each category is summarized to have most useful insights. It covers malware research with various datasets available including Android apps based datasets. It has brief discussion on ML and deep learning methods along with their methodology. The insights of this paper provide good understanding on different methods existing, their approach, datasets collected and used besides evaluation metrics. These insights along with the proposed framework and experimental results can trigger further research with specific possibilities in future. Since the dataset utilized in this paper has more number of attributes the survey has identified that deep learning based approaches has more efficiency rather than the ML approaches. The integration of deep learning approaches with optimization techniques has enhanced the utilization of resources.

**Keywords:** *Malicious Code Detection, Malware Detection, Machine Learning, Deep Learning, Malware Detection Optimization*

## 1. INTRODUCTION

With the evolution of fast app development many untrusted sources are releasing their apps similar to real apps. It is difficult to identify the difference between real and clone apps. Clone apps attack or controls the personal information in the mobile with the help of malware virus.

Malware is the term which indicates some sort of malicious piece of software that causes damage to computers, network and communication systems. It is written for disruption of computing infrastructure and deny users of services. Malware causes many issues such as system crash, down of network, server may be down, reduce speed of Operating System (OS), loss of disk space and even

disruption of an important software causing services come to a halt. Given the proliferation of malware on the Internet, malware detection is essential since it serves as a computer's early warning system for malware and cyberattacks. It stops hackers from accessing the computer and guards against data breaches. The process of checking a computer and its contents for malware is known as malware detection. Because it uses a variety of methods and techniques, it is efficient at identifying malware. It's a two-way process; in fact, it's rather complicated. The great news is that malware identification and uninstallation only take a few minutes.

The learning approaches scan the attributes of the training dataset and analyze them to check its characteristics. The dataset contains

vital information about the source of the server, its complete domain listing, trust issues with apk, and much other information. For the efficient working of the model, learning algorithms can utilize the optimality principle to reduce the number of attributes and can enhance the execution of algorithms.

With the advent of AI based techniques associated with data science, it is now possible to predict malware and prevent its adverse effects. As explored in [1], [2], [3], [4], to specify few researches, there has been ML methods that are able to learn from historical events and monitor malicious events and detect presence of malware. In the same fashion, as discussed in [11], [12], [13], to mention few, there have been different methods that are associated with deep learning. There are certain optimization methods that use some sort of evolutionary methods used to optimize prediction performance. Chaung and Wang [2] proposed a hybrid fusion model using fusion logic that combines both normal and malicious behavioral models based on machine learning. Their model performance better than the existing models like SVM but suffers from data imbalance problem.

The research carried out in this paper has witnessed tremendous possibilities in terms of detecting malware and make intelligent response systems. Kakavand *et al.* [8] applied SVM and KNN for detecting malware and training is given based on the manifest keywords. Dynamic approach and hybrid malware detection approach is missing in their models. Lopes *et al.* [9] focused on mobile malware and investigated on different ML approaches including permissions based, API calls based, permissions and used features based and permissions and API calls-based ones. It has problem with small data sets and data imbalance problem. Wang *et al.* [13] proposed a deep learning framework known as Droid Deep Learner for Android malware detection. Their method analyzes Java source code and also manifest files for obtaining API function calls and permissions in order to get all features from Android apps. Chander, N. *et al.* [45] proposed Metaheuristic feature selection with deep learning enabled cascaded recurrent neural network for anomaly detection in Industrial IoT Environment. Kumar,

M.U. *et al.* [46] proposed Dependable Solutions Design by Agile Modeled Layered Security Architectures. Shrivani, D. *et al.* [47] proposed Designing Dependable Web Services Security Architecture Solutions. Krishna Prasad, A.V. *et al.* [48] proposed Designing Dependable Business Intelligence Solutions Using Agile Web Services Mining Architectures, Mahalakshmi *et al.* [49] proposed Automatic Water Level Detection Using IoT.

Then these features are subjected to deep learning to detect presence of malware. From the literature, it is understood that the possibilities to control malware towards cyber security is very affirmative. It has given sufficient hypotheses to look into this research further. The Artificial Intelligence (AI) based methods in terms of ML and deep learning techniques paved way for improving prediction performance. In this regard, it is very significant to explore such methods for efficient malware detection. This paper's contributions are:

1. An academic investigation into the present state of the art on ML and deep learning methods for automatic malware detection.
2. Model have provided summary of findings associated with ML methods, deep learning methods and optimization approaches.
3. Model also provide possible research gaps.

The remainder of the paper is structured as follows. Section 2 reviews literature on different ML approaches for malware detection. In the same fashion, Section 3 reviews existing methods of deep learning for malware detection automatically. Section 4 on the hand focuses on the optimization methods including evolutionary approaches to improve prediction performance. Section 5 presents the research gaps that could trigger further research. Section 6 draws conclusions.

## 2. MACHINE LEARNING FOR MALWARE DETECTION

Machine learning techniques are widely used for malware detection. In order to foresee or make judgments without being explicitly told to, machine learning algorithms create a model using

sample data, often known as training examples. Numerous techniques employ machine learning methodologies. Without explicit instructions, machine learning programmes can complete tasks. To learn how to perform particular tasks, computers use the data that is readily available. It is feasible to create algorithms that teach a machine how to complete all the steps required to solve the problem at hand for simple tasks; the machine doesn't need to learn how to do these things. A machine learning system uses the prediction models it has created from historical data to anticipate the result when it gets new data. How well the output is predicted depends on the amount of information employed; a larger data collection makes it simpler to build a model that forecasts the result more precisely. supervised learning, unsupervised learning, and reinforcement learning are the three different forms of machine learning.

Mahindru and Singh [1] investigated on Android malware detection using ML approaches. They extracted 123 permissions from Android applications used Random Forest (RF), Decision Tree (DT) and Naïve Bayes (NB). Their approach is to collect features from. APK files and train ML classifiers to detect malicious apps. Homodelver, the dataset can be updated with new apps so as to improve performance. Chaung and Wang [2] proposed a hybrid fusion model using fusion logic that combines both normal and malicious behavioral models based on machine learning. Their model performance better than the existing models like SVM but suffers from data imbalance problem. Liu *et al.* [3] made a review of several approaches found in the literature for malware detection. They contributed in the reliability estimation of different models for a comparative study. Lou *et al.* [4] defined a model where sensitive data flows and topics are used with ML for Android malware detection. Homodelver, their method needs further improvement with extraction of dynamic information flows. Ma *et al.* [5] proposed a hybrid model that combines control flow graphs and ML techniques along with LSTM to detect Android malware. Homodelver, finding malicious API position in code and finding category of malwares are yet to be explored.

Zhao *et al.* [6] investigated on the samples used for malware detection in Android apps. They studied the issues associated with sample duplication and identified duplication types in order to handle them model in the empirical study. Jung *et al.* [7] used ML techniques like SVM and useful API calls to discover Android malware. They intended to combine SVM and Artificial Neural Network (ANN) in future for better performance. Kakavand *et al.* [8] applied SVM and KNN for detecting malware and training is given based on the manifest keywords. Dynamic approach and hybrid malware detection approach is missing in their models. Lopes *et al.* [9] focused on mobile malware and investigated on different ML approaches including permissions based, API calls based, permissions and used features based and permissions and API calls based ones. It has problem with small data sets and data imbalance problem.

Gavrilut *et al.* [24] used ML approaches for detection of malware. In the process, they investigated on the tradeoff between model accuracy and memory footprint. Takawale and Thakur [25] proposed a methodology based on ML techniques. It includes extracting permissions from Android apps, feature extraction, model building and model evaluation. Homodelver, their research needs further improvement with larger datasets. Data is collected by gaining permission to access "Canadian Institute of cyber security's datasets". Bayazit *et al.* [26] explored different methodologies based on ML approaches towards malware detection. Kosmidis *et al.* [27] a methodology for feature engineering and modeling a classifier towards malware detection. The malware binary data is converted to 8-bit vectors and it is converted grayscale image in order to perform malware detection process further. Different ML approaches are employed with comparative study. Ali *et al.* [30] investigated on Android apps and the malware problem with different kinds of methods.

Arvind Mahindru *et al.* [41] have proposed an MLDroid framework to detect malware in Android devices using four distinct machine learning techniques. In the model to detect malware, different phrases have been chosen. Initially, upload the .apk package and extract the permissions from the device, carried by feature

selection. Xinning Wang *et al.*[42] developed a Multi-dimensional kernel feature and featured weight-based detection, identifying malware and benign apps. To identify whether the device is malware affected or not, the kernel feature initially uploads the data.

Table 1: Summary of important ML approaches used for malware detection

S. No & Author	Journal, Year & Ref	Methodology	Datasets/Tools	Metrics	Drawbacks
1. Mahindra and Singh	ACM, 2017 [1]	Machine Learning with RF, DT and NB (application is specific to Android malware)	Dynamic Permission Dataset	Precision, Recall, Accuracy and F1-score	Needs to consider more recent apps in the dataset
2. Chuang and Wang	IEEE, 2015 [2]	Hybrid methodology that fuses normal and malicious behavioral models	Android .APK files datasets	Precision, Recall and Accuracy	Data imbalance needs to be taken care of.
3. Lou <i>et al.</i>	IEEE, 2019 [4]	Sensitive data flows and topics are used with ML for Android malware detection	Derbin dataset	TP, TN and Accuracy	Extraction of dynamic information flows is still desired.
4. Ma <i>et al.</i>	IEEE, 2019 [5]	A hybrid of ML techniques	droZo and AMD	Precision, Recall,	Finding malicious

		ues and LSTM		Accuracy and F1-score	API position in code and finding category of malware are yet to be explored.
5. Arvind Mahindru <i>et al</i>	Neural Computing and Applications 2020[41]	MLDroid-framework	Virus Total dataset	Accuracy, F-measure	soft computing models to attain better detection rate for malware
6. Xinning Wang <i>et al</i>	Neurocomputing 2021[42]	Kernel feature-based framework and WBD	Malware & Benign dataset	TPR, TNR, Accuracy	the issue of overfitting

Table 1 shows the summary of important ML approaches found in the literature. The methods are diversified with various approaches.

### 3. DEEP LEARNING FOR MALWARE DETECTION

Deep learning models do have depth in learning process and essentially meant for improving accuracy in prediction. Kim *et al.* [11] proposed a methodology for Android malware detection using multi-model neural net approach with representative feature selection. It could reap benefits of multiple deep neural networks. Homodelver, in their work, the dynamic features addition is still desired. Hou *et al.* [12] investigated

on the prediction based on Linux kernel system call graphs using deep learning based autoencoders that contain encoding and decoding procedures in order to find anomalies. In their method, random event generation in component traversal method is yet to be done. Wang *et al.* [13] defined a framework known as DroidDeepLearner for Android malware detection. Their method analyzes Java source code and also manifest files for obtaining API function calls and permissions in order to *get all* features from Android apps. Then these features are subjected to deep learning to detect presence of malware. In future, they intended to employ fuzzy association rule mining on the feature set. Li Dongfang *et al.* [14] proposed a fine-grained deep neural network approach towards Android malware detection. Homodelver, their method does not collect dynamic features from Android applications. Su *et al.* [15] explored on Android apps dataset collected from VirusTotal with feature extraction, feature learning and deep learning. Homodelver, their deep learning method needs further improvement for better performance.

Zegzhda *et al.* [16] explored different methods in deep learning along with observation of sequence of API calls to detect Android malware. Zero pixel padding in their method causes more overhead to the system. Sahbadiya *et al.* [17] used deep learning model for automatic feature extraction and make decisions on the malware presence and also the family of malware. They intended to improved their model to detect malware in Android apps even before they are installed. Zhu *et al.* [19] focused on the sensitive data usage of Android apps illegally. They proposed a deep learning based method for detecting it. It has feature extraction and feature granularity determination for improving performance. Sandeep [21] used static analysis of Android apps using deep learning for detecting presence of malware. Different activation functions and optimizers are used for performance evaluation. He and Kim [32] created malware images and used CNN based deep learning approach for malware detection. CNN based deep learning models with malware images are used for empirical study [40]. Homodelver, to be effective in their research, more pre-trained models are to be evaluated. Karbab *et al.* [33] proposed a methodology deep learning

based neural network for automatic detection of malware. It has novelty in using many sources of data for more efficient feature extraction. The datasets used are Malgenome dataset, Drebin dataset, Google Play and MalDozer dataset.

Hardy *et al.* [34] exploited deep learning technique with optimizations. Their method takes Portable Executable (PE) files and obtains API calls that are used to train deep autoencoders. The deep autoencoder with encoding and decoding procedures is crucial for detection of abnormalities and identify malware presence eventually. Alzaylaee *et al.* [35] proposed a methodology with dynamic analysis from phone devices, feature ranking and deep learning for classification. Zhong and Gu *et al.* [37] proposed a multi-level approach in deep learning for malware detection. They used a tree structure to combine many deep learning models for improving performance. Ye *et al.* [38] proposed deep learning framework for malware detection. Their method combines both Boltzmann mechanics and deep stacked autoencoder. PI files are taken as inputs and Windows API calls are used for training the system. Different feature representations and modeling is intended for their future endeavor. Yuxin and Siyi *et al.* [39] used deep belief networks (DBNs) for detection of malware. PE files are used for feature extraction and then the deep model is trained with extracted features. They intend to find tradeoff between unlabeled data size and performance.

Nan Zhang *et al.* [43] discovered a deep learning method to analyze malware-affected devices as known feature selection is the most commonly used technique for retrieving data. By overcoming the feature here, the author used NLP to recognize text. Avowed the processes initially data should be generated then the reverse engineering comes to phrase to provide an integrated document. These documents are trained to text, and the data is distributed as effect and unaffected files. It needs to be compared to know whether the algorithm has achieved the best result. Therefore, the classification of various techniques is not better than the proposed algorithm. By avoiding feature selection, it is easy to auto-detection. Here the algorithm considers only text formatted data.

Table 2: Summary of important deep learning based malware detection approaches

S. No & Author	Journal, Year & Ref	Methodology	Datasets/Tools	Metrics	Drawbacks	and Kim	2019 [32]	based deep learning models with malware images	collected from Andro-dumpsys project of Korea University	and FPR	pre-trained models are to be evaluated.
1. Kim <i>et al.</i>	IEEE, 2018 [11]	Multi-model neural network approach with representative feature selection	Virus Share and Malgenome project	Accuracy	Dynamic feature additions is still desired.	6. Karb ab <i>et al.</i>	Elsevier (Digital Investigation), 2018 [33]	Deep learning based neural network for automatic detection of malware	Malgenome dataset, Drebin dataset, Google Play and MalDozer dataset	Precision, Recall, Accuracy and F1-score	Pre-trained deep learning models needs to be exploited.
2. Hou <i>et al.</i>	IEEE, 2016 [12]	Prediction is based on call graphs of Linux kernel system	Android malware dataset is collected from Comodo Cloud Security Center	Accuracy	Random event generation in component traversal method is yet to be done.	7. Hardy <i>et al.</i>	Data Mining, CSREA [34]	Deep learning for PE based API calls extraction and modeling	Dataset is provided by Comodo Cloud Security Center	TPR and FPR	Autoencoder with sparsity constraints is yet to be explore
3. Su <i>et al.</i>	IEEE, 2016 [15]	Feature extraction, feature learning and deep learning	Dataset is collected from VirusTotal	Precision, Recall, F1-score and overall accuracy	Deep learning method needs further improvement for better performance.	8. Alzaylaee <i>et al.</i>	Computer and Security, Elsevier [35]	Dynamic analysis from phone devices, feature ranking and deep learning for classification	McAfee Labs dataset	Precision, Recall, Accuracy and F1-score	Self-adaptation kind of IDS is expected to be done
4. Feng <i>et al.</i>	IEEE, 2020 [18]	Used CNN and autoencoder with two-layer deep learning approach	CICAnd Mal2017 and Virus Total	Precision, Recall and Accuracy	The two levels of detection work is to be integrated for leveraging performance	9. Xiao <i>et al.</i>	Hindawi [10]	A novel approach "behavior-based deep learning framework	VX Heaven for malware samples and Cuckoo Sandbox for benign	Precision, Recall, Accuracy and F1-score	SAE along with other deep learning models are yet to be
5. He	IEEE,	CNN	Dataset is	TPR	More						

		ork (BDLF) " with behavio ral graphs	samples		studied
10. Nan Zhan g <i>et al</i>	Applied Soft Comput ing [43]	Deep Learn ing, TC- Droid	Drebin Datasets	Accu racy, Peci sion, Recal l, and F1- score	Only text conten t is identif ied.

Fatima *et al.* [22] proposed a deep learning based methodology focusing on APK reverse engineering, Genetic Algorithm (GA) for feature selection with discrimination capability. Limited data size is the limitation of their work. Irshad *et al.* [23] focused on feature optimization using GA. Feature optimization is made for efficient runtime analysis using GA. Limited dataset and data imbalance problems are yet to be resolved in their work.

Table 3: Summary of important optimization approaches used for malware detection

Table 2 shows the summary of important deep learning approaches found in the literature. The methods are diversified with various kinds of modus operandi towards efficient detection of malware.

#### 4. OPTIMIZATION TECHNIQUES FOR MALWARE DETECTION

Optimization algorithms and evolutionary methods are widely used along with ML and deep learning models for performance improvement in detection process. The optimization technique is a potent instrument for obtaining the ideal operating circumstances and the required design parameters. This would serve as a guide for the experimental work and lower the risk and operating costs. Finding the decision variable values that optimize one or more intended objectives is referred to as optimization. The design of objective functions and the choice of optimization techniques determine the dependability of optimal solutions. A numerical method that explains and forecasts the behavior of the process is necessary for optimization. Optimization search may be used to estimate independent variables in complicated non-linear processes. Dynamic processes' uncertainty variables might be identified using robust optimization. Designing multiphase reactors and flow systems and scaling up methodologies could both benefit from optimization. Manufacturing and engineering operations won't be as effective as they are currently without optimization of design and operations.

S. No & Auth or	Journ al, Year & Ref	Method ology	Dataset s/ Tools	Metri cs	Drawba cks
1, Fatim a <i>et al.</i>	IEEE, 2019 [22]	APK reverse engineer ing, Genetic Algorith m (GA) for fea ture se lectio n with discrim ination capabilit y	Android apps dataset collecte d from C3i Center of IIT Kanpur	Sensiti vity, specifi city, ROC and AUC	Limited dataset size.
2. Irsha d <i>et al.</i>	IEEE, 2019 [23]	Feature optimiz ation is made for ef ficient runtime analysis using GA.	Dataset is colle cted fr om And roid ap ps in window s and AV-Test Engine antivirus	Sensiti vity, specifi city, accura cy and AUC	Limited dataset and data imbalan ce prob lem
3.Raz ak <i>et al.</i>	Arab J Sci Eng, Sprin ger [29]	Bio inspired methods to optim ize fea tures to improve predicti on perform	Drebin dataset and Google Play	Precisi on, Recall , Accur acy and F1- score	Still more training samples are de sired for le veragi ng per form ance

		ance			
4.Cai <i>et al.</i>	Comp uters & Securi ty, Elsevi er [31]	Joint optimiz ation and feature modelig hting and modelig ht-mappin g	Drebin, AMD, Google Play and APKPure.com	Precisi on, Recall , Accur acy and F1- score	Correlat ion betmod elen features is required for further improve ment.
5. Yang <i>et al.</i>	IEEE, 2019 [36]	CNN model with optimiz ations	Google play and AMD dataset	Precisi on, Recall , Accur acy and F1- score	Extra features and control flows are to be incorpo rated.
6. Christ iana <i>et al.</i>	iJIM, 2020 [40]	Ensemb le learning based optimiz ation	Android malware dataset	Sensiti vity, specifi city, precisi on, recall, F- measu re and ROC.	Need further optimiz ation with appropri ate bio- inspired method.

Table 3 shows the summary of important optimization approaches. The methods are having different kinds of optimizations towards efficient detection of malware.

## 5. RESEARCH GAPS

The existing researches found in the literature, there have been rich set of methods that used ML approaches with different datasets and different methodologies. An important problem found is that there is need for more recent apps in the dataset as explored in [1]. It is also important to see that there is balance in the data. Imbalanced data [2] is a problem while dealing with supervised learning. As discussed in [3], it is important to extraction of dynamic information flows or topics or permissions associated with mobile apps is to be considered. In case of deep learning methods also it is understood that dynamic features addition is essential [11]. As discussed in [18], [32] and [33] it

is understood that there is still need for improving deep learning methods such as integration of underlying mechanisms, usage of more pre-trained models and usage of autoencoders with hybrid models. In essence, it is understood that there is need for further research on the ML and deep learning methods for detecting Android malware. There is also a need for exploiting optimization approaches to leverage predictive performance.

From the literature review, it is ascertained that efficient and timely detection of malware is an important problem considered. Therefore, automatic detection of malware is a challenging problem considered.

With these research gaps, the paper has formulated the problem statement as follows: The proposed model should design an efficient dimensionality reduction technique using optimization techniques. The model needs to reduce the execution time as well as resources so instead of traditional approaches it has to choose blending or stacking techniques which predicts the class labels with majority voting. The model needs to design a counter attack for the malware detected apps.

## 6. PROPOSED FRAMEWORK AND RESULTS

The important hypothesis considered from the review of literature is that "ML and deep learning methods have the capability to improve efficiency in malware detection". The preliminary solution is provided in this paper in the form of a framework that enabled malware detection to be done with several ML models and CNN with deep learning.

### 6.1 The Framework

The proposed framework is based on the supervised learning. It is presented in Figure 1. Drebin dataset [50] is used for empirical study. The Drebin dataset includes a text file for each application. All of the application's properties are described in the text file. Each asset falls into one of eight categories (S1 to S8). The frequency with which each category appears in the text file can therefore be used as a feature vector for such application. A feature vector for one application might be 1-12-0-3-5-67-9-4, for instance. Therefore, the S1 category appears once, the S2 category twelve times, and so forth. The DREBIN collection comprises up to 545,333

behavioral features and 123,453 large sample sizes for Android applications, including 5560 malicious samples.

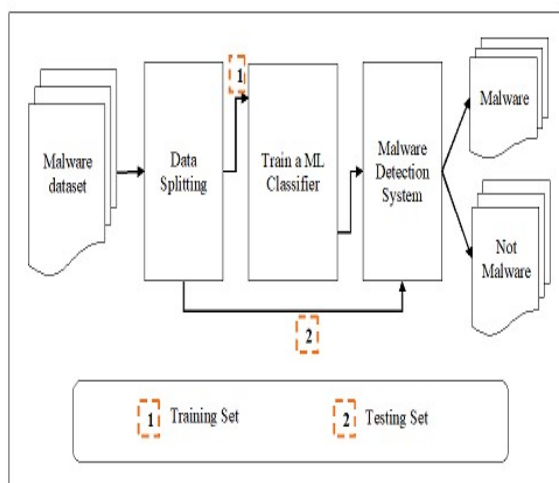


Figure 1: Proposed framework

As presented in Figure 1, the proposed framework splits data into training and testing with 80:20 ratio. Different prediction models are trained to have malware prediction model. After completion of training process, the ML models gain required intelligence to form a malware detection system that takes testing data and performs a malware detection process resulting in the classification of malware samples discriminated from genuine instances. The test data is evaluated to detect malware based on the knowhow of trained model. The prediction models are directly taken from [51] and evaluated the dataset aforementioned.

The Naive Bayes algorithm is a Bayes theorem-based supervised learning technique for classification problems. It often employs a huge training set to categorise texts. The Naive Bayes Classifier is one of the easiest and most effective classifiers currently in use. Making efficient machine learning models that can make precise predictions is made easier by this method. Because it is a probabilistic classifier, it creates an approximation based on the likelihood that an object will show up. Examples of Naive Bayes method applications include spam filtering, sentiment analysis, and article classification.

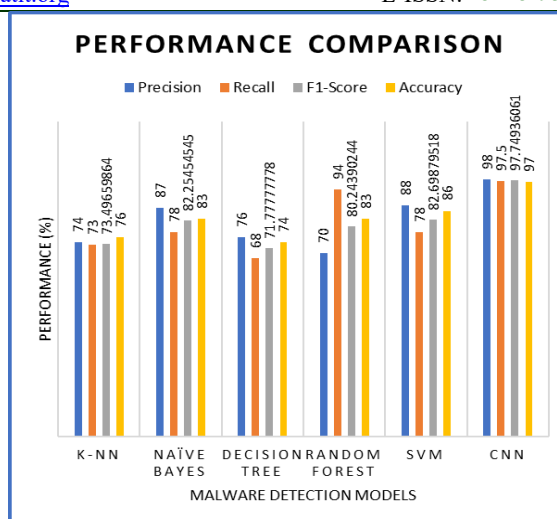


Figure 2: Performance of malware detection models

As presented in Figure 2, the performance of the malware detection models is presented. The highest performance of machine learning models is 86% which is exhibited by SVM. When compared with ML models, the CNN, deep learning model, showed better performance with 97%. When the results of CNN are compared with state of the art, it is observed that the deep learning model used in [21] showed 95.57% accuracy. Based on this observation, CNN used in this paper showed better performance. The existing models utilized traditional approaches with good accuracy but most of them have not diagnosed the bias and variance problems. So, the proposed model utilizes the genetic approaches with dynamic parameter setting model to update the positions in the global spectrum and reduces the dimensions. It also tries to combine activation functions to customize the layers with new functions so that the neurons can travel towards the destination class with more speed and less learning rate.

## 7. CONCLUSION AND FUTURE WORK

In this paper, the model has made a review of literature for ascertaining the current academic thinking and the methods being used or methods possible to detect malware automatically. The study in this paper is divided into three categories known as ML methods for malware detection, deep learning methods for malware detection and optimization methods for improving malware prediction performance. Each category is summarized to have the most useful insights. It covers malware research with various datasets

available including Android apps based datasets. It has brief discussion on ML and deep learning methods along with their methodology. The insights of this paper provide good understanding on different methods existing, their approach, datasets collected and used besides evaluation metrics. These insights can trigger further research with specific possibilities in future. Particularly, in future, model intend to propose a ML framework for Android malware detection to leverage prediction performance and improve the state of the art. Another direction is to propose more efficient deep learning framework for improving malware detection significantly.

## REFERENCES

- [1] Mahindru, Arvind, Singh, Paramvir, "Dynamic Permissions based Android Malware Detection using Machine Learning Techniques" *Proceedings of the 10th Innovations in Software Engineering Conference, ACM Press the 10th Innovations in Software Engineering Conference*, (India), February 05-07, 2017, pp.202–210.
- [2] Hsin-Yu Chuang and Sheng-De Wang, "Machine learning based hybrid behavior models for Android malware analysis", *IEEE International Conference on Software Quality, Reliability and Security*, 2015.
- [3] Kaijun, Xu Shengmodeli, Xu Guoai, Zhang, Miao Sun, Damodeli; Liu, Haifeng, "A Review of Android Malware Detection Approaches based on Machine Learning", *IEEE Access*, 2020, pp.1–30.
- [4] Lou Songhao Cheng, Shaoyin Huang, Jingjing Jiang Fan, "TFDroid: Android Malware Detection by Topics and Sensitive Data Flows Using Machine Learning Technique", *IEEE 2nd International Conference on Information and Computer Technologies (ICICT)*, (Kahului, HI, USA), March 14-17, 2019, pp.30–36.
- [5] Ma Zhuo, Ge Haoran, Liu Yang, Zhao Meng, Ma, Jianfeng. "A Combination Method for Android Malware Detection Based on Control Flow Graphs and Machine Learning Algorithms", *IEEE Access*, 2019, pp.1–11.
- [6] Zhao Y, Li L, Wang H, Cai H, Bissyandé, T.F. Klein, J.Grundy, "On the Impact of Sample Duplication in Machine-Learning-Based Android Malware Detection", *ACM Transactions on Software Engineering and Methodology*, March -30, 2021, pp.1–38.
- [7] Jung Jaemin, Kim Hyunjin, Shin Dongjin, Lee Myeonggeon, Lee Hyunja, Cho Seong-je, Suh Kyoungwon, "Android Malware Detection Based on Useful API Calls and Machine Learning", *IEEE First International Conference on Artificial Intelligence and Knowledge Engineering (AIKE)*, (Laguna Hills, CA, USA), September 26-28, 2018, pp.175–178.
- [8] Kakavand Mohsen, Dabbagh Mohammad, Dehghantanha Ali, "Application of Machine Learning Algorithms for Android Malware Detection", *Proceedings of the 2018 International Conference on Computational Intelligence and Intelligent Systems - CIIS* (Phuket, Thailand), November 17-19, 2018, pp.32–36.
- [9] Lopes Joao, Serrao Carlos, Nunes Luis, Almeida Ana, Oliveira Joao, "Overview of machine learning methods for Android malware identification", *IEEE 7th International Symposium on Digital Forensics and Security (ISDFS)*, (Barcelos, Portugal), June 10-12, 2019, pp.1–6.
- [10] AeGuen Kim, BooJoong Kang, Mina Rho, Sakir Sezer and Eul Gyu Im. "A Multimodal Deep Learning Method for Android Malware Detection using Various Features", *IEEE Access*, 2017, pp.1-16.
- [11] Kim TaeGuen, Kang Boo Joong, RhoMina, Sezer Sakir, ImEul Gyu, "A Multimodal Deep Learning Method for Android Malware Detection using Various Features", *IEEE Transactions on Information Forensics and Security*, 2018, pp.1–16.
- [12] Hou Shifu, Saas Aaron, Chen Lifei, Ye Yanfang, "Deep4MalDroid: A Deep Learning Framework for Android Malware Detection Based on Linux Kernel System Call Graphs", *IEEE/WIC/ACM International Conference on Modelb Intelligence Workshops (WIW)*, (Omaha, NE, USA), October 13-16, 2016, pp.104–111.
- [13] Wang Zi, Cai Juecong, Cheng Sihua, Li Modelnjia, "DroidDeepLearner: Identifying Android malware using deep learning", *IEEE 37th Sarnoff Symposium* (Newark, NJ, USA), September 19-21, pp.160–165.
- [14] Li Dongfang, Wang Zhaoguo, Xue Yibo, "Fine-grained Android Malware Detection

- based on Deep Learning” *IEEE Conference on Communications and Network Security (CNS)*, (Beijing, China), May 30 – June 01, 2018, pp.1–2.
- [15] Su Xin, Zhang Dafang, Li Modelnjia, Zhao Kai, “A Deep Learning Approach to Android Malware Feature Learning and Detection”, *IEEE Trustcom/BigDataSE/ISPA*, (Tianjin, China), August 23-26, 2016, pp.244–251.
- [16] Zegzhda Peter, Zegzhda Dmitry, Pavlenko Evgeny, Ignatev Gleb, “Applying deep learning techniques for Android malware detection”, *Proceedings of the 11th International Conference on Security of Information and Networks – SIN*, ACM Press the 11th International Conference, (Cardiff, United Kingdom), September 10-12, 2018, pp.1–8.
- [17] Sabhadiya Sagar, Barad Jaydeep, Gheewala Jaydeep, “Android Malware Detection using Deep Learning”, *IEEE 3rd International Conference on Trends in Electronics and Informatics (ICOEI)*, (Tirunelveli, India), April 23-25, 2019, pp.1254–1260.
- [18] Feng Jiayin, Shen Limin, Chen Zhen, Wang Yuying, Li Hui, “A Two-Layer Deep Learning Method for Android Malware Detection Using Network Traffic”, *IEEE Access*, 2020, Vol 8, pp.125786–125796.
- [19] Dali Zhu, Hao Jin, Ying Yang, Wu, Di Modeliyi Chen, “DeepFlow: Deep learning-based malware detection by mining Android application for abnormal usage of sensitive data”, *IEEE Symposium on Computers and Communications (ISCC)*, (Heraklion, Greece), July 03-06, 2017, pp.438–443.
- [20] M. Sujithra, G. Padmavathi, “Enhanced Permission Based Malware Detection in Mobile Devices Using Optimized Random Forest Classifier with PSO-GA”, *Sciences, Engineering and Technology*, 2016, Vol. 21 (7), pp. 1-10.
- [21] HR, Sandeep, “Static Analysis of Android Malware Detection using Deep Learning”, *IEEE International Conference on Intelligent Computing and Control Systems (ICCS)*, (Madurai, India), May 15-17, 2019, pp.841–845.
- [22] Fatima Anam, Maurya Ritesh, Dutta Malay Kishore, Burget Radim, Masek Jan, “Android Malware Detection Using Genetic Algorithm based Optimized Feature Selection and Machine Learning”, *IEEE 42nd International Conference on Telecommunications and Signal Processing (TSP)*, (Budapest, Hungary), July 01-03, 2019, pp.220–223.
- [23] Irshad Areeba, Maurya Ritesh, Dutta, Malay Kishore, Burget Radim and Uher, Vaclav, “Feature Optimization for Run Time Analysis of Malware in Windows Operating System using Machine Learning Approach”, *IEEE 42nd International Conference on Telecommunications and Signal Processing (TSP)*, (Budapest, Hungary), July 01-03, 2019, pp.255–260.
- [24] Gavrilut Dragos Teodor, Anton Dan Gabriel and Popoiu, George, “Machine Learning based Malware Detection. How to Balance Memory Footprint with Model Accuracy”, *IEEE 19th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)*, (Timisoara), September 21-24, 2017, pp.232–238.
- [25] Takawale Harshvardhan C, Thakur Abhishek, “Talos App: On-Device Machine Learning Using TensorFlow to Detect Android Malware”, *IEEE 5th International Conference on Internet of Things: Systems, Management and Security (IoTSMS)*, (Valencia, Spain), October 15-18, 2018, pp.250–255.
- [26] Bayazit Esra Calik, Koray Sahingoz, Ozgur and Dogan Buket, “Malware Detection in Android Systems with Traditional Machine Learning Models: A Survey”, *IEEE 2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*, (Ankara, Turkey), June 26-28, 2020, pp.1–8.
- [27] Kosmidis, Konstantinos and Kalloniatis, Christos, “Machine Learning and Images for Malware Detection and Classification”, *Proceedings of the 21st Pan-Hellenic Conference on Informatics, ACM Press the 21st Pan-Hellenic Conference*, (Larissa, Greece), September 28-30, 2017, pp.1–6.
- [28] Mohd Faizal Ab Razak, Nor Badrul Anuar, Fazidah Othman, Ahmad Firdaus, Firdaus Afifi and Rosli Salleh, “Bio-inspired for Features Optimization and Malware Detection”, *Computer Engineering and Computer Science*, 2017, pp.1-17.
- [29] Anam Fatima, Ritesh Maurya, Malay Kishore Dutta, Radim Burget and Jan Masek, “Android Malware Detection Using Genetic Algorithm based Optimized Feature

- Selection and Machine Learning”, *IEEE Access*, 2019, pp.1-4.
- [30] Ali Mariam Al, Svetinovic, Davor, Aung Zeyar and Lukman, Suryani, “Malware detection in android mobile platform using machine learning algorithms”, *IEEE 2017 International Conference on Infocom Technologies and Unmanned Systems (Trends and Future Directions) (ICTUS)*, (Dubai, United Arab Emirates), December 18-20, 2017, pp.763–768.
- [31] Lingru Cai , Yao Li and Zhi Xiong, “Android Malware Detection Based on Feature Modelighting with Joint Optimization of Modelight-Mapping and Classifier Parameters”, *Computers & Security*, 2020, pp.1-21.
- [32] Ke He and Dong Seong Kim, “Malware Detection with Malware Images using Deep Learning Techniques”, *IEEE International Conference On Trust, Security And Privacy In Computing*, 2019, pp.1-8
- [33] ElMouatez Billah Kabab, Mourad Debbabi, Abdelouahid Derhab and Djedjiga Mouheeb, “Automatic framework for android malware detection using deep learning”, *Digital Investigation*, 2018, pp.48-59.
- [34] William Hardy, Lingmodeli Chen, Shifu Hou, Yanfang Ye and Xin Li, “A Deep Learning Framework for Intelligent Malware Detection”, *Data Mining*, 2016, pp.1-7.
- [35] Mohammed K. Alzaylaee, Suleiman Y. Yerimab and Sakir Sezerc, “Deep Learning Based Android Malware Detection Using Real Devices”, *Computers & Security*, 2019, pp.1-28.
- [36] Shao Yang, “An Image-inspired and CNN-based Android Malware Detection Approach”, *IEEE Access*, 2019, pp.1-3.
- [37] Modeli Zhong, Feng Gu. “A Multi-Level Deep Learning System for Malware Detection”, *Accepted Manuscript*, 2019, pp.1-40.
- [38] Yanfang Ye, Lingmodeli Chen, Shifu Hou, William Hardy and Xin Li, “DeepAM: a heterogeneous deep learning framework for intelligent malware detection”, *Springer-Knowl Inf Syst*, 2017, pp.1-21.
- [39] Ding Yuxin and Zhu Siyi, “Malware detection based on deep learning algorithm”, *Springer Neural Comput & Applic.*, 2017, pp. 1-12
- [40] Abikoye Oluwakemi Christiana, Benjamin Aruwa Gyunka, “Optimizing Android Malware Detection Via Ensemble Learning”, *IEEE Access*, 2020, Vol. 14 (9), pp.1-18.
- [41] Mahindru, A., & Sangal, A. L., “MLDroid—Framework for Android malware detection using machine learning techniques”, *Neural Computing and Applications*, 2020.
- [42] Xinning Wang, & Chong Li, “Android malware detection through machine learning on kernel task structures”, *Neurocomputing*, 2021,435, pp.126–150
- [43] Nan Zhang, Yu-an Tan, Chen Yang, Yuanzhang Li, “Deep learning feature exploration for Android malware detection”. *Applied Soft Computing*, 2021, pp.1-7.
- [44] Daniel Arp, Michael Spreitzenbarth, Malte Huebner, Hugo Gascon, and Konrad Rieck "Drebin: Efficient and Explainable Detection of Android Malware in Your Pocket", *21th Annual Network and Distributed System Security Symposium (NDSS)*, February 2014.
- [45] Chander, N., Upendra Kumar, M. “Metaheuristic feature selection with deep learning enabled cascaded recurrent neural network for anomaly detection in Industrial Internet of Things environment”, *Cluster Computing*, 2022.
- [46] Kumar, M.U., Kumar, D.S., Rani, B.P., Rao, K.V., Prasad, A.V.K., Shravani, D. “Dependable Solutions Design by Agile Modeled Layered Security Architectures”, *Advances in Computer Science and Information Technology. Networks and Communications. CCSIT 2012. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, 2012 vol 84. Springer, Berlin, Heidelberg.
- [47] Shravani, D., Suresh Varma, P., Padmaja Rani, B., Upendra Kumar, M., Krishna Prasad, A.V. “Designing Dependable Web Services Security Architecture Solutions”, *Ad-vances in Network Security and Applications. CNSA 2011. Communications in Com-puter and Information Science*, 2011 vol 196. Springer, Berlin, Heidelberg.
- [48] Krishna Prasad, A.V., Ramakrishna, S., Padmaja Rani, B., Upendra Kumar, M., Shravani, D, “Designing Dependable Business Intelligence Solutions Using Agile Web Services Mining Architectures”, *Information Technology and Mobile Communication. AIM 2011. Communications in Computer and*

- 
- Information Science*, 2011 vol 147. Springer, Berlin, Heidelberg.
- [49] Mahalakshmi, C.V.S.S., Mridula, B., Shravani, D, “Automatic Water Level Detection Using IoT”, *Advances in Decision Sciences, Image Processing, Security and Computer Vi-sion. Learning and Analytics in Intelligent Systems*,2020, vol 4. Springer, Cham.
- [50] Drebin Dataset. Retrieved from [https://github.com/chamarh/drebin\\_dataset](https://github.com/chamarh/drebin_dataset)
- [51] Scikit-Learn, *Machine Learning Library*. Retrieved from <https://scikit-learn.org/stable/index.html>