Mesopotamian journal of Cybersecurity
Vol.2023, **pp**. 134–157
DOI: https://doi.org/10.58496/MJCS/2023/017; ISSN: 2958-6542
https://mesopotamian.press/journals/index.php/CyberSecurity

Research Article

Machine Learning-Based Detection of Smartphone Malware: Challenges and Solutions

Amneh Alamleh [1,*], Sattam Almatarneh [2], Ghassan Samara[3], Mohammad Rasmi [4]

[1,2] *Department of Data Science and Artificial Intelligence, Faculty of Information Technology, Zarqa University, Zarqa 13100, Jordan*

[3] *Department of Computer Science, Faculty of Information Technology, Zarqa University, Zarqa 13100, Jordan*

[4] *Department of Cyber Security, Faculty of Information Technology, Zarqa U'niversity, Zarqa 13100, Jordan*

ARTICLE INFO

Article History

Received 21 April 2023
Accepted 10 July 2023
Published 10 Aug. 2023

Keywords

Smartphone Security

Malware Evaluation and Benchmarking

Machine Learning

Multi-Criteria Decision Making (MCDM)

ABSTRACT

The goal of this research is to review the researcher's different attempts with respect to new and emerging technology in malware detection techniques based on machine learning approaches over smartphones. The aim is to evaluate and benchmark these techniques, identify the current landscape of research in this area, and construct a cohesive taxonomy. The available options and gaps will be analyzed to provide valuable insights for researchers regarding the technological environments within this research area. A deep analysis review was conducted to identify studies addressing smartphone security based on machine learning approaches in order to identify all related articles. The outcomes of the last classification scheme of these articles were categorized into types of detection: dynamic analysis, static analysis, hybrid analysis, and uniform resource locator (URL) analysis. The evaluation criteria used in malware detection techniques, with respect to machine learning approaches for smartphones, include accuracy, precision rates (including true positive, false positive, true negative, false negative), training time, f-measure, detection time, area under the curve, true positive, true negative, false positive, false negative, and error rate. Additionally, our classification covers the main machine learning techniques used in the reviewed studies. The taxonomy includes three distinct layers, each reflecting one aspect of the analysis. We also reviewed the details of various types of malicious and benign datasets used within malware detection. Furthermore, open issues and challenges were identified in terms of evaluation and benchmarking, which jeopardize the utilization of this technology. We have described a new recommendation pathway solution that aims to enhance the measurement process of smartphone security applications.

1. INTRODUCTION

Smartphones serve as phones and Portable personal computers which enable its users with various types of services including internet browsing, social networking, email, short message service (SMS), not to mention other important services like maps, Global Positioning System (GPS), and mobile payment applications [1, 2]. The Internet world is constantly expanding, and an increasing number of different individuals, corporations and groups largely rely on network and its resources for different purposes. In light of this increasing demand, smartphones rose to be a significance contributor to the ease of many lives which cannot be replaced, especially for work and leisure, the penetration level of these technology devices shows a remarkable growth. The cybercrime world notices the large number of vulnerabilities associated with the expanding of the usage of the mobile device, the development of specific worms and spyware software. The area of cyber security considering its nature had to keep changing to meet the expectations of users and deal with different kind of vulnerabilities that either are new of previously identified from this phenomenon, issues that are mainly associated with the mobile environment nature, especially when it comes to important data protection and privacy. For example, a high-profile data leakage in 2014 has affected many people around the globe, among the most affected ones are an American celebrity that was due to the weak accounts passwords they implemented and were linked with their Apple devices which in turn were exploited to retrieve private images of those celebrities. In 2015, the Hacking Team Company involved with the data stated that government entities could install malware on targeted smartphones for unauthorized purposes including

*Corresponding author. Email: mr.maad.alnaimiy@baghdadcollege.edu.iq

eavesdropping activities and data theft. During the same period of time, due to the significance expansion of mobile networks along with internet integration, many users increased the access point's ubiquity through the network. This feature is derived also from the increasingly available bandwidth and other technologies including 4G/LTE, thus providing high-speed connections. Therefore, mobile devices constitute new targets of attacks and effective resources for implementing and executing the attacks themselves because of their huge and continuously increasing number [3, 4]. Lately the attention of a large number of software developers was drawn to the development area of malware detection approaches. Large number of different technologies are currently utilized in the malware detection [5]. Nevertheless, this research concentrated machine learning techniques and their usages. Moreover, using machine learning techniques for malware detection became common and widely used to gain more accurate results and increase the smartphone security [6, 7]. Although, all the benefits obtained from these malware detections based on machine learning; however, the challenging that will be faced is how to choose malware detection applications, which can produce results that are accurate, not to mention ensuring that the results are also highest in term of performance between different available alternatives [8]. The large diversity among available machine learning techniques which commonly used for malware detection makes it difficult for deciding on which of them to use in detection. Thus, the challenge is associated with how the valuation and comparison of malware detections to choose the best one, particularly when there is no dedicated technique that is far better than the other ones, in addition majority of these technique suffer from accuracy lack and computational efficiency [9-11], [12]. On the other hand, the difficult part is associated with the evaluation and comparison because of the multiple evaluation criteria and conflict between them [8, 13-15]. The evaluation and benchmarking procedure for the detection of malware for the smartphone security is critical during the quest of acquiring technique that can produce best results. A similar process is essential since there will be a cost to the users for the selection of malware detection which could result in losing the personal and private information. In order to select malware detection techniques from many available techniques, there is a requirement for both evaluating and benchmarking processes to guarantee the best selection, especially since these techniques are not cheap as well as related with the privacy and security users which are used the smartphone [7]. However, two main drawbacks face developers in malware detection area-based machine learning approach over smartphone. Firstly, how malware detection techniques are capable of performing evaluation of multi-criteria. After that, is how this malware detection technique is benchmarked as opposed to other existing techniques? In the current literature, both specific areas, the first one is evaluation and second one is benchmarking. In addition, they are considered both a challenge and a gap. This paper's aim is to shade lights on research efforts with respect to emerging and new technology of malware detection – based on machine learning approach over smartphone in evaluation and benchmarking with the aim of mapping the related studies into coherent taxonomy, and to highlights the challenge and open issues in with respect to the evaluation and the benchmarking, finally, proposes the recommended solution in order to dealing the identified challenge and issues. The remains of this study made up of seven parts. Part 2 reviews and deeply analyze previous studies. As for Part 3, it shows the distribution of evaluation criteria and machine learning techniques used in the literature. Part 4 discusses the datasets were used in the reviewed articles. Part 5 presents a discussion on challenges, in addition to open issues which are associated with evaluation and benchmarking for malware detection-based machine learning techniques. Part 6 provides the recommended solution. Part 7 presents the methodology of the proposed solution. Finally, conclusion was presented in part 8.

2. TAXONOMY ANALYSIS

This study's goal is to shade and highlight the most common criteria used by various researchers with respect to evaluation for malware detection techniques based on machine learning approach over smartphones; map the landscape of research from the literature towards a coherent taxonomy of crossover amongst three layers. Figure 1 shows the crossover taxonomy used in order to review the related articles; the first layer (middle) shows types of malware detection techniques, namely, dynamic, static, hybrid and URL; the second layer (right) shows different machine learning used in detection techniques over smartphones; the third layer (left) show multiple criteria for evaluation malware detection techniques. The following sections describe the crossover amongst three layers. Such sections present the developed systems for malware detection techniques based on various machine learning approach over smartphones, which used multiple criteria for evaluation malware detection techniques. Several articles performed a detection procedure with examination steps for evaluation, which are considered detection techniques. The articles are divided into four types of detection mechanisms namely, dynamic, static, hybrid and URL as illustrated in the literature. The below sections present descriptions of each technique and the articles included, as well as present the description of the crossover amongst three layers.

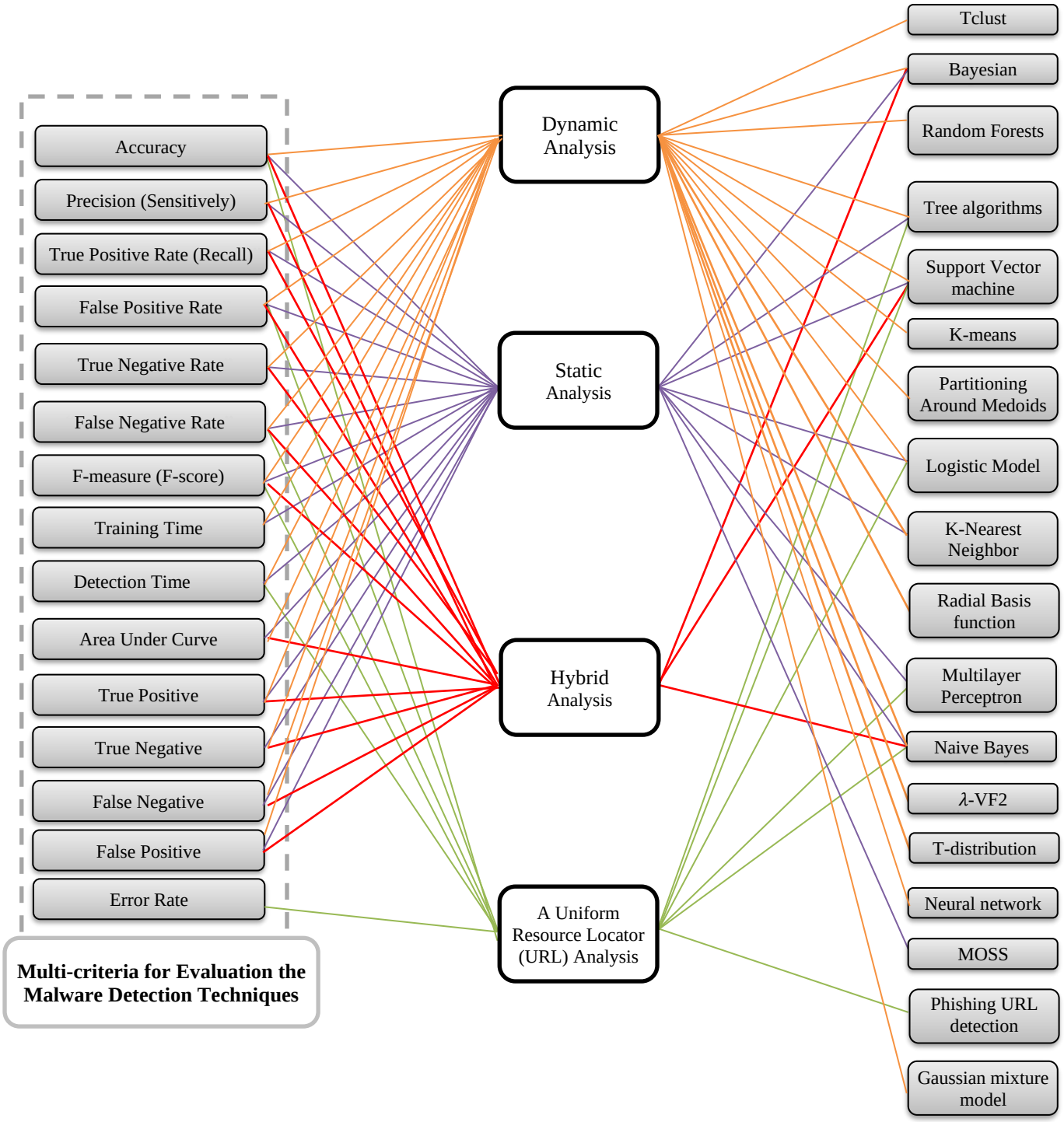

Fig. 1. Crossover taxonomy for malware detection techniques based on machine learning approach over smartphones.

2.1 Dynamic Analysis

This technique studies the application behavior after installation on smartphone and then decides whether the application is benign or malware. Many topics, such as the anomaly or behavior-based detection techniques to study the smartphone behavior after the application is installed, are included in this section. After analyzing the behavior, these methods can assess the application status as malicious or benign. Numerous authors in the literature provided schemes based on behavior analysis by utilizing various techniques of machine learning based on their study which is associated with evaluation used

criteria. For the trimming methods that are based on trimmed techniques of Kmeans and Tclust . The techniques will be used towards identifying the homogenous groups of applications which demonstrates similar behavior according to three criteria. The criteria are accuracy, rate for true positive and rate for false positive [16]. Another study by [17] has utilized seven criteria. The criteria are accuracy, rate for true positive, false positive, time of training, true positive, false negative and false positive, towards evaluating the efficient framework based on Bayesian method , the Bayesian method which is developed with the aim of analyzing the traffic behavioral changes in real time of Android Apps [17]. Authors in [18] adopted accuracy, precision, f-measure and false positive to evaluate a novel machine learning dynamic analysis approach. The approach goes by the name "Label Conditional Mondrian Inductive Conformal Prediction" or "LCMICP" with Random Forest classifier. It cab present provably valid confidence guarantees for each malware detection. Another study used six criteria, namely, accuracy, sensitively, recall, false positive rate, f-measure and training time to evaluate a new Android malware method with dynamic detection that is based on service matrices for call co-occurrence, as well as this study used machine learning techniques including KNN, Decision Tree, Random Forest, Naive Bayes, Logistic regression, in addition to Support Vector to classify and verify the feature sequence of an Android Apps whether it can expose Android malware behaviors or not [13]. The study [8] included nine criteria, namely, accuracy, precision, recall, false positive rate, f-score, true negative, true positive, false negative and false positive towards evaluating linear support vector machine (SVM)-Based dynamic Android Malware Detection. In addition the study also aimed to compare the performance of SVM malware detection along with that of other machine learning classifiers. They established that in order to detect malware effectively in the Android platform with monitored resources during runtime of the application. The study [19] demonstrated that statistical techniques for mining can be vulnerable to attacks which can cause a random smartphone malware behavior, so this study analyzed real-time collections of smartphone usage statistics and detected malware programs based on different classifiers such as logistic regression, decision tree, artificial neural network, support vector machine, and naive Bayes. This study adopted four criteria including accuracy, true positive rate, area under curve, false positive rate in evaluating the results for algorithms used in analysis. The [20] study utilized technique of supervised classifier of Random Forest on an Android feature dataset. The reason is to measure the accuracy of Random Forest and classify Android application behavior in the same time, in order to classify malicious or benign applications, three criteria, namely, accuracy, false positive and false negative were used to evaluate the results for class algorithm used in classification. Another study [21] utilized three criteria, namely, accuracy, false positive rate and true negative rate to evaluate the results of an improved Bayesian classification method that developed to analyze android application behaviours. The study [14] focused on the use of transport gestures with the aim of preventing the issue of misuse for three major smartphone capabilities. The capabilities are the calling service of the phone, the NFC reading feature and camera. The authors have expresses that the three gestures are dynamically detectable with high overall accuracy, in addition to being distinguished from one another and other activities including the malicious or benign ones, which will act as defense against viable malware. Moreover, this study used seven criteria including accuracy, f-measure, true negative, true positive, false negative, false positive and precision to evaluate different classifiers, namely, Random Forest (RF), Logistic Model Trees (LMT), Logistics (L), Random Tree (RT), Naive Bayes (NB), Simple Logistic (SL) across different sensors subset that would result in best accuracy. The study [22] evaluate the results of the software behavior-based anomaly detection system by using four evaluation criteria, namely, accuracy, recall, false positive rate and detection time. The power consumption anomalies were presented by the authors, in addition to others including the temperature of the battery and the traffic of the network with the use of data for three different algorithms, namely, Random Forest (RF), Support Vector Machine (SVM) and logistic model trees (LMT). The study [23] presented a new technique of establishing dynamic birthmarks by proposed Android dynamic detection system Demadroid which resists the obfuscated attack based on λ-VF2 algorithm. Multiple criteria have been used to evaluate the proposed technique such as, accuracy, sensitively, recall, false positive rate and false negative rate. Another study [15] used two linear classification algorithms, namely, Naïve Bayes and Logistic Regression in the automatic malware detection via a latent network behavior analysis based on the results from sandbox. This study used accuracy, precision, recall, true positive, true negative, false positive and false negative to to evaluate the detection capability of proposed method. The study [24] used the accuracy only to evaluate the efficacy of the developed system for intrusion detection based on Bayes Classifying, such system applied to determine whether there is an invasion on the smartphones through analyzing the abnormalities of Android system are dynamic and are able to locate software that is malicious, along with state monitoring of the system for intrusion detection system which monitors the process and flow of network in smartphone. Another study [25] developed a malware behavior-based detection system. The system has the capability to examine calls of the system in order to capture the runtime behavior for the software, in addition to applying Support Vector Machine (SVM) and Naive Bayes (NB). Both are used in order to learn the dynamic behavior of software execution. The authors evaluated their system based on three criteria including accuracy, training and time detection time. The study [26] used Decision Tree algorithm, the study presents a new detection system for anomaly which is behavior-based, the system has the detection ability for identifying meaningful deviations in the network of mobile application's behavior. Moreover, detection time, true positive rate and false positive rate criteria were used in this study

to evaluate the results of the new behavior detection system. The study [27] proposed a new behaviour-based approach in order to estimate the abnormality's rate of the usage smartphone, it's based on several machine learning approaches such as K-means, Partitioning Around Medoids) PAM (, Gaussian mixture model) GMM(, t-distribution, Trimmed K-means, Tclust. This study evaluated the result of the proposed approach by using accuracy, true positive rate, false positive rate and area under curve criteria. Three machine learning techniques were used in the study [28], namely, logistic regression (LR), artificial neural network (NN) and support vector machine (SVM) to mine behavioural data logs in a novel mobile malware detection system. In addition, f-measure, accuracy, sensitively and recall criteria were used to evaluate the results in this study. A classification component was introduced by [29], it aims to identify Android Apps with the use of traffic flow analysis and detect behavioral changes in real time. This study used multiple criteria to evaluate the classification component The components include different elements including the accuracy and recall, in addition to the precision, rate for false positive, true positive, false negative, true negative and f-measures. The author in [30] only utilized rate for false positive and rate for true positive criteria with the aim of evaluating fully-fledged tool. The toos which he evaluated was based on different techniques including K- Nearest Neighbor (KNN), Bayesian Networks, Random Forest and Radial Basis function (RBF). Those techniques have the capabilities to dynamically analyse any iOS software when it comes to invocation method. In addition, they also are capable of producing exploitable results. These results are able to be utilized with the aim of automatically tracing software's behavior or it also can manually distinguish whether the software has malicious code or not.

2.2 Static Analysis:

This type of detection technique analyzes the source code or checks the application file permissions before the installation on smartphone. Many authors in the literature used this type of detection technique, as shown in the next topics: Starting with an effective Bayesian classification models-based method. This study shade lights on the proposed approach effectiveness based on multiple evaluation criteria. The criteria are accuracy, under curve, recall, precision, false positive rate, true negative rate, in addition to false negative rate [31]. A novel framework that is based on three models of machine learning. The machine learning models are Naïve Bayes, Random Forest and Multilayer perception to analysis code of potential Android malware apps statically by extracting the intention and their permission requests. Multi-evaluation criteria were utilised to evaluate the proposed framework, namely, accuracy, recall, true positive, true negative, false positive and false negative [32]. The study [33] used four criteria including area under curve, accuracy, precision and true positive rate to evaluate three machine learning approaches. The approaches are Support Vector Machines (SVM) followed by Decision tree and Bagging method, that us utilized in order to perform simple static analysis, is uses permission and API calls in order to find system functions associated with each App and detect malicious Android Apps. The study [7] proposed an approach based Naïve Bayes, Simple logistic, decision tree and random tree, Random forest, which used ensemble learning for Android malware static detection, the authors evaluated the proposed approach based on multi-criteria such as, accuracy, sensitively, true positive rate, false positive rate, true negative rate and false negative rate. Another study [34] proposed multiple static feature-based mechanism to better detection is acquired by training five methods of machine learning, namely, Decision tree (J48), in addition to other including Naive Bayes (NB), Decision stump (DS), Support Vector machine (SVM), and Random tree (RT), and merging their decisions with the use of collaborative approach that is based on probability theory. This model evaluated based on multiple criteria such as f-measure, area under curve, accuracy, precision and true positive rate. The study [1] proposed an Android malware detecting system based k-nearest neighbour (KNN) algorithm and static dataflow analysis related API-level features, that is capable of accurately identifying Android malware, not to mention efficiently discovering sensitive data transmission paths. Eight evaluation criteria were utilized to evaluate the proposed system, namely, accuracy, sensitively, true positive rate, false positive rate, true positive, true negative, false positive and false negative. Another study [10] utilized criteria such as, detection time, accuracy, true positive, true negative, false positive and false negative criteria to evaluate the results of the proposed method, this method is based on (MOSS) algorithm to detect variants of known malware families in Android devices with the use of simplify Dalvik instructions. The study [35] proposed a method for detection with the use of multiple classifier system on the basis of support vector machine (SVM). Every base classifier is responsible for one type of malware. The authors used multiple criteria to evaluate the proposed method such as detection time, accuracy, recall, false positive rate, true positive, true negative, false positive and false negative. The study [11] presented detection system that is a permission-based on Android malware that is based of logistic regression technique, APK Auditor that classify Android Apps as malicious or not by using static analysis. The authors evaluated the efficiently of their work through multi-criteria such as true positive rate, false positive rate, accuracy, true positive, true negative, false positive and false negative. The study [36] used only accuracy criteria to evaluate the results of the proposed framework, which has capabilities to extract various static features types from every application with static analysis. In addition, it can employ the ensemble for multiple classifiers. These classifiers include K-Nearest Neighbor (K-NN), Naive Bayes (NB), Support Vector Machine (SVM), Classification, Regression Tree (CART) and Random Forest (RF), which aim to detect malware applications and

make a category for the benign applications. In another study by [37], the author also adopted only accuracy to evaluate the proposed KUAFUDET, which is a learning enhancing defense system based on number of machine learning methods including random forest (RF), K-nearest neighbor (KNN) and support vector machine (SVM). In addition, this system has an adversarial detection capability that performs training phase offline to select and extract contributing features, these features are taken from the training set for preprocessing purpose. Furthermore, it has the online detection phase that use classifiers which was trained by the first phase. Study [38] suggested a structural analysis that is based on Naïve Bayesian (NB), in addition to other like Support Vector Machine (SVM) and lastly the Reduced error pruning tree (REPTree) in order to classify either botnet and benign apps . This stage is done with the use of characteristics of botnet related and unique patterns for the requested permission, in addition to the used features. The author has evaluated the results by utilizing various evaluation criteria including f-measure, training time, accuracy, precision, rate for both true positive and false positive.

2.3 Hybrid Analysis:

The authors proposed the combined techniques of static and dynamic to overcome the weaknesses of the two techniques and provide robust detection technique for smartphone against malware. The study [39] presented a smartphone dual defence protection framework based Random forest and J.48 techniques, which involves the verification server which utilizes the call statistics of system with the aim of identifying possible malicious applications, in case the software is not infected and clean, after that the app will be released to the designated markets. On the other hand, users who run the application has the ability to invoke the traffic of the network monitoring tool with the aim of analyzing its traffic, in addition to determining if network characteristics matches the observed ones from malware applications This study utilized many criteria like accuracy, recall, false positive rate, and false positive and false negative for evaluating the proposed framework. In Another study by [40] adopted six criteria. The criteria are accuracy, false positive rate, recall, false negative rate, true positive and false negative towards evaluating the results for a detection system which is both novel and hybrid which is based on a new open-source framework; CuckooDroid, the system enables the utilization of features in Cuckoo Sandbox's with the aim of analyzing Android malware with the use of dynamic and static analysis by utilizing support vector machine (SVM) as a classifier. Another study by [41] the author used multiple-criteria like accuracy, false negative rate, recall, sensitively, false positive and false negative towards evaluating the proposed MARVIN system efficiency. The system has the ability to leverage linear classifiers and a Support Vector machine Technique by combining Static and dynamic analysis towards assessing the linked risk with unidentified and unknown Android Applications as malicious score. An automatic malware detection system was developed in study [9], this developed system was based on Support Vector machine classifier. The system utilizes mix of code features for the static analysis with the runtime behavioural analysis patterns. This study evaluated the results with the use of multi-criteria like accuracy, true negative and true positive. The last study [42] utilized Naive Bayesian in the proposed methodology, it has the capability to easily acquire the Android Apps information with no need to use any type of complex code analyzing technique. After that, our risk score is based in the behaviour which is expected from a benign app, in addition an alert will be sent to the user with respect to the abnormal request for permissions and not only exclusive to the known malware. The results are evaluated by authors based on multiple criteria. These criteria include the under-curve area, the rate for all the following: true positive, false positive, true negative and the f-measures.

2.4 A Uniform Resource Locator (URL) Analysis:

Another type of malware detection techniques over smartphones is URL Analysis Web application services are accessible via personal mobile device using the internet, two ways this process can be done, by either typing the URL in the web browser search area or by clicking a link which will automatically link to the web application. Users are installing application from one of three choices, either by downloading from the official store, a third part app store or from an APK file downloaded from other websites. In any of the previous cases, the URLs operates as ways that obtain access to a web application. In addition, it makes it an exploitable tool that is utilized by attackers to infect their malwares into the victim's devices. This security approach concentrated on the protection of users from applications which could pose a risk in mobile web browser to spread malware. The study [43] used two criteria including accuracy and detection time to evaluate the results of the scalable mobile URL classifier system based Logistic Regression, Decision Forest, Naive Bayes and, in order to detect malicious websites accurately and in timely manner with very minimal overhead on the smartphones. The study [44] developed a methodology based on seven machine learning methods; Multilayer Perceptron (MLP), Random Forest (RF), Support Vector Machines (SVMs), logistic regression (LR), Naïve Bayes (NB), and C4.5, which is tool for anti-phishing URL in order to block a phishing attack. It can perform the blocking with two ways, the first is to mask the possible phishing URL, and the second is to alert the user about the potential threat. The authors in this study evaluated the results using multiple-criteria such as, accuracy, error rate, and false positive and false negative rates. The study [45] described a lightweight approach based J48 decision tree algorithm, which can classify malicious web pages with the use

of URL lexical analysis alone, several criteria were used to results evaluation of the proposed approach including accuracy, false positive rate, F1-Score, detection time. The last study [46] utilised only accuracy criteria to evaluate the results of the multi-Classification for Malicious URL Based on Improved Co-Forest algorithm, which construct URL multi-classification model.

3. DISTRIBUTION RESULTS

3.1 Distribution of Evaluation Criteria

Figure 2 illustrates the various evaluation criteria distribution used in various reviewed studies, namely, accuracy, error rate, area under curve, f-measure, precision, false positive, true negative, false negative, true positive, detection time, training time, true negative rate, false negative rate, false positive rate, true positive rate.

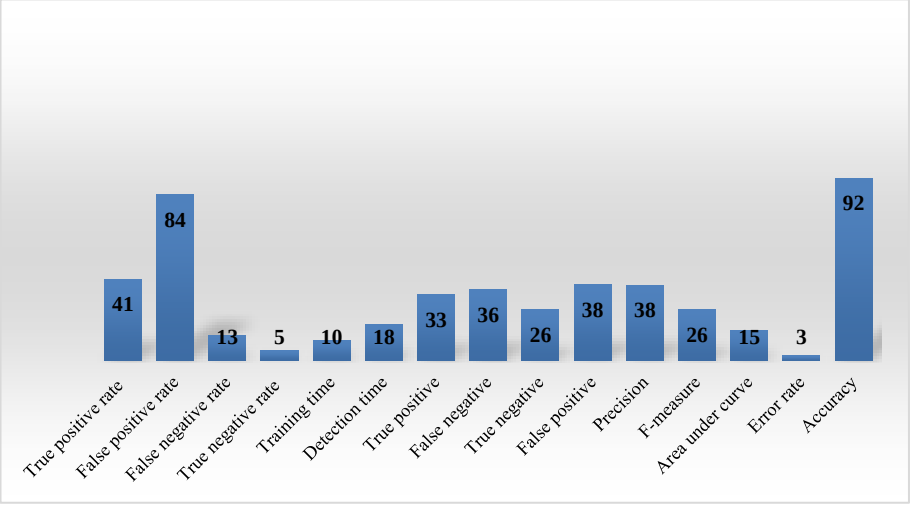

Fig. 2 Distribution of evaluation criteria in the taxonomy

Last figure 2, displays varying criteria's ratio that are used in various studies through subsections of titles in the taxonomy. These studies have used the criteria for evaluation purposes on their techniques and methods. Majority of the studies 92% is depend on accuracy criterion in evaluating of malware detection; then 84% of studies depend on false positive rate and 41% true positive rate; false positive and Precision is used in 38% of studies; 36% of studies are depend on false negative; true positive, False negative, F-measure, detection time, area under curve, false negative rate, training time, true negative rate, Error rate were used in 33%, 26%, 26%, 18%, 15%, 13%, 10%, 5%, 3% of studies respectively. Regarding to our analysis, most of studies used multiple evaluation criteria to measure the quality and performance the malware detection techniques which used. This study indicated that each study used one or number of evaluation criteria that proper with their target of developed malware detector, as well as the different ratios of used in figure 2 shows that variation in usage. This case of evaluation criteria usage will be challenge if we want to comparing among some of malware detection to select the best of them.

3.2 Distribution of machine learning techniques

Figure 3 illustrates the different machine learning techniques which used in various reviewed studies; most of study used more than one technique.

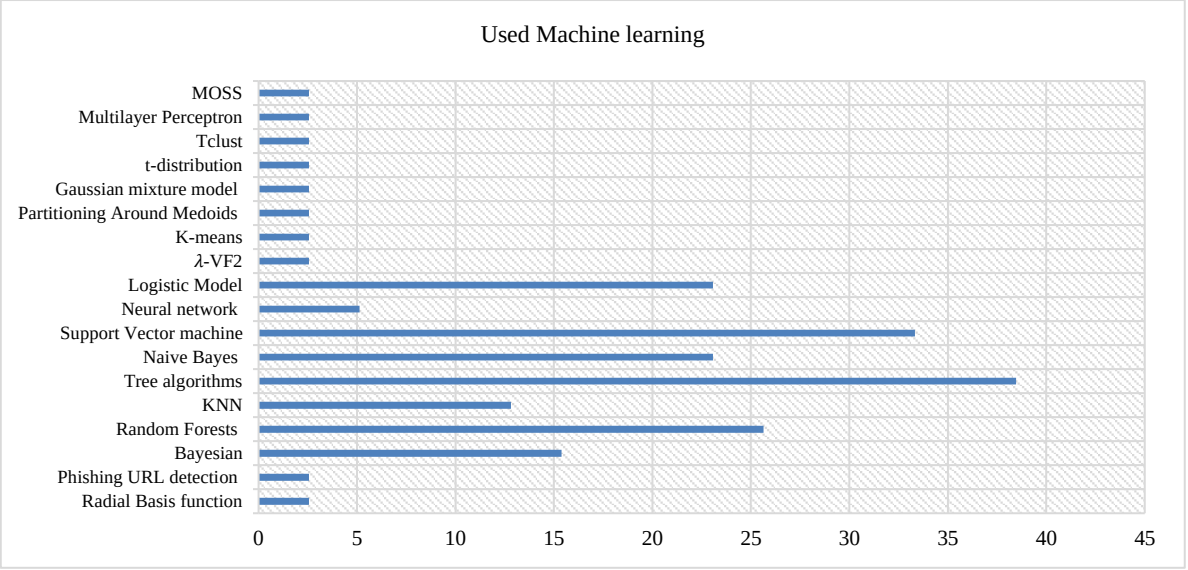

Fig. 3 Ratio of usage of various machine learning techniques in reviewed studies

According to figure 3, tree algorithms, support vector machine, were most used in reviewed studies, then follow them random forests, naive Bayes and logistic model algorithms. Each of remaining algorithms is used in 3% of reviewed studies.

4. DATASETS AVAILABLE USED

In our review, the authors are reviewed the details of various types of malicious and benign datasets used within malware detection and classification over smartphone studies. The details of such datasets in our review are summarised in Table 1. Details such as dataset description with total number of malicious and benign applications used within different malware detection techniques including dynamic, static, hybrid and URL, different static and dynamic features that extracted from datasets and source of datasets were included.

Table 1 Datasets used in the literature

No	Ref	Dataset description	Source
Datasets used within dynamic detection technique			
1	[16]	-Smartphone behavioral application dataset which contains 20% of abnormal apps. -Datasets which has two different types of malicious intents, some of the malicious intents include the simultaneously launch of many processes to over the CPU of the device, as for the second type, it's the Denial of Service (DOS) attack.	Obtained From [47]
2	[17]	-Two approaches existed, the first one is for imbalanced dataset, and the second approach is aimed for balanced dataset. The process is done by taking an equal number of samples for each class (App). -Data collection process in the midst of real network setup which performs on multiple periods and it includes different measurements aimed for the same features	Real datasets collected through interacting users with multiple Apps
3	[18]	-Massive dataset which is created by collecting data of the installation of 1866 malicious application and 4816 benign applications over real android smartphones. -The data recorded includes different number of permission information including information of the Binder, CPU, Memory, Battery and network	The dataset is created by means of installation for android application files (.apk) on LG E400 Android device then recording the device state during the running of the application and simulating the interaction of the user
4	[13]	- 1000 normal samples - 15 malicious code taken from different families	Normal Sample are Available and gathered from Google Play on the following link (https://play.google.com/store?hl=in)

		- 50 cases from Each Family, totally 750 are labelled as malicious samples	The Following Code Contains the Malicious code families that were gathered from AndroMalShare (http://andromalshare.androidmalware:8080/#.com)
5	[8]	The data set is composed of 90% normal and 10% malicious applications	Not mentioned
6	[48]	Over the entire dataset there are feature vectors which are from 20 application that are benign, 6 malware programs (Results are 120 feature vectors per application), and 30 randomized profiles for malware programs (Every profile of the randomized ones varies from the other with respect to the randomization amount), in the total of 56 applications.	Application for Feature extraction was developed for collecting the data
7	[20]	- The authors provided dataset based on observations for 407 benign samples and 1330 malicious apk samples. - The original dataset had a total of 32342 data, feature vector samples with 7535 benign samples classified as positive class and 24807 malicious samples classified as negative class.	Obtained from https://github.com/VT-Magnum-Research/antimalware
8	[21]	477 application samples	Not mentioned
9	[14]	Data was collected using an app, the collection of the data was extensively for gestures. These gestures include call, tap and snap, as well as different control activities.	There were a total of 23 users in data collection
10	[22]	49 malware variants and 200 samples for malware, in addition to 200 Top Free benign application were the components of the samples	-The sample applications of malware are taken from Malware Genome project as in the following link (www.malgenomeproject.org) -The benign applications are downloaded from Google Play for Indonesia from 200 Top free applications as in the following link (https://play.google.com/store?hl=in)
11	[23]	This reference uses two kinds of datasets in experiments, simulative malicious samples, and real malware samples (22 ADRD and 16 Bgserv).	- Constructed 10 simulative samples. The malicious codes in these samples are basically the same
12	[15]	-Until the publish time of this reference, 310 Android Malware types appeared. -102 unique malware samples were identified and then Tested in Droid Box.	This reference used the same dataset from [49]
13	[24]	-Collected 15 malicious application and 45 normal applications -15 random application was selected from every experiment from the aforementioned applications (i.e., 45 normal applications and 15 malicious applications).	The source of the dataset is the Android market
14	[25]	-Real world malware sample and benign applications were collected -System calls were gather by this reference with the use of a tool called Strace, the tool has the ability to record the system calls	The source of the dataset is the Android market
15	[26]	-16 dataset were extracted for the purpose of calibration experiment, in addition to preparing them from 8 applications. -Records for training and testing were main components of each o the used 16 datasets	Data collected for 8 applications including twitter, gmail, facebook, groupme, twitter, firefox, whatsapp, and linkedin.
16	[27]	--A malware with capabilities of launching background processes was used with the aim of overloading the CPU of the device. -Normal and abnormal cases were identified for DOS attack by the author -20% malicious apps were used with thousands of benign apps.	Malware developed in [47]
17	[28]	The intrusion activities which were deployed over Android OS are created by means of modifying the external metasploit library. This will enable the investigation of Rapid7 Vulnerabilities. Device activities can record activities like calls history, SMS messages, browser, intents and sampling of process and connections into raw logs.	Not mentioned
18	[29]	Different number of datasets which features 44,921 benign applications, in addition to 6,154 malicious ones. For example, Game, Personalization and weather. Sensitive API has used 6 terms in order to understand the malicious and benign datasets distribution	- Malicious apps from VirusShare (http://virusshare.com/) - Benign apps from Google Play (https://play.google.com/), and Anzhi Market (http://www.anzhi.com/)
19	[30]	The exploitation of unofficial iOS frameworks creates new malware instances	Not mentioned

		Several of previous malicious subroutines which were contained in the current malware that were not considered were used Malicious subroutines contained in existing malware.	
Datasets used within static detection technique			
20	[31]	2000 application were collected, the 2000 APKs have thousand malware samples taken from 48 different families, also 1000 benign applications	Downloaded Benign applications from official and third-party markets for Android.
21	[32]	- The collection of reliable data source is possible by the three different sets of Android applications (APK files) - Samples for Malware creates the first dataset, the dataset has 1260 Android malware APK samples from 49 different malware families - The collection of the second and the third datasets are enabled by Benign applications	- The first dataset from [50] - The second and third from Google Play (https://play.google.com)
22	[33]	- 1250 benign Android APK files. - 49 different malware families have 610 malware samples	From [50]
23	[7]	6863 applications in Total, 2925 of these applications were malware and 3938 applications were benign.	Obtained from McAfee's internal repository in the following link (https://www.mcafee.com/)
24	[34]	- API calls and Permissions are extracted from 904 clean applications and 1073 malicious files. - 135 permissions in total, in addition to 210 API calls which have been extracted for constructing the two feature sets.	Malware Geome Project as in the following link (http://www.malgenomeproject.org)
25	[1]	- More than 2,200 real-world Android apps were used - Including 1,160 benign apps and 1,050 malwares.	VirusShare is the source from which the malicious samples were collected as in the following link (http://virusshare.com/) This project is an online well-known project for malware repository. Different applications categories were gathered which is covered by the benign samples. It was collected from a popular Chinese android application market as in the following link (http://sj.qq.com/myapp/4)
26	[10]	- 3000 applications from sample library as analysis data set were selected by this reference. - 1000 known malicious applications and 2000 known normal applications are included.	- This reference used samples taken from the virus database of North Carolina State University, the large sample base is a collection of reptiles and filtered normal application in laboratory. - Collecting sources are Google Play as in the following links (https://play.google.com/), Baidu (http://pcappstore.baidu.com/en/index.php), Tencent (https://www.tencent.com/)
27	[35]	- 370 Android applications are included in the dataset, in addition it include 139 Android malicious applications and 231 Android benign applications. - 334 features in Total are extracted: 305 and 29 are related to uses-permission and uses-feature respectively.	Malicious applications can be downloaded from the Contagio mobile as in the following link (http://contagiominidump.blogspot.com) - The benign applications are selected the top 600 application from Anzhi Market as in the following link (http://www.anzhi.coml)
28	[11]	- total of 8762 applications were collected and analyzed - total of 6909 malwares were collected - total of 1853 benign applications	- malware repository collected from a named contagio mobile as in the link (http://contagiodump.blogspot.com), Drebin (http://user.informatik.uni-goettingen.de/~darp/drebin/), dataset and Android Malware Genome Project (http://www.malgenomeproject.org) And [50] - Benign applications were downloaded from Play Store as in the link (https://play.google.com/)
29	[36]	- Large data set containing 107,327 benign applications were used and 8701 malicious apps for testing purposes. - Each APK file Extract 2,374,340 features	- The benign apps were crawled from one of the biggest app markets in China called Anzhi (http://www.anzhi.com). - Malware apps were collected in wild
30	[37]	- The first large collection of 252,900 Android application samples. - Including 242,500 benign applications, and the other 10,400 malicious APK files	- downloaded Benign apps from Google Play Store as in the link (https://play.google.com/). - 1260 malicious APK files were validated in [50] and the remaining are downloaded from Contagio Mobile

			Website as in the link (340 APKs) (http://contagiominidump.blogspot.com/), Pwnzen Infotech Inc. (4500 APKs) (http://www.pwnzen.com/), and (4300 APKs) from [51].
31	[38]	- A large collection of datasets from different sources is analysed towards the understanding of the botnet C&C structure, characteristics and attacks, that are utalized to extract the requested permissions and used features. - Various 122176 of benign applications were collected. - They have collected publicly 9756 android botnet applications	- Different categories of applications that are benign are gathered from many open-source sites like Google Android market as in the links (https://play.google.com/), SlideMe (http://slideme.org/), and Pandaapp (http://www.pandaapp.org/) - The different categories of android botnet applications are gathered from Android Malware Gnome project as in the link (http://www.malgenomeproject.org), [51-54]
Datasets used within hybrid detection technique			
32	[39]	1260 Android malware samples in 49 different malware families.	From [50]
33	[40]	- The final dataset contains 6000 benign applications and 5560 malware samples features. - The authors extracted 190,367 different static and dynamic features from datasets.	- To collect benign apps, the authors designed crawler and craw a large number of apps in China app stores, as in the link http://www.appchina.com, http://www.as.baidu.com, http://www.mm.10086.cn. - The used malware samples in experiment are acquired from [51].
34	[41]	- More than 135000 Android applications and 15000 malware samples. - They extracted 496,943 different features (154,939 dynamic analysis features and 342,004 static analysis features).	- The collected the benign apps from the Google Play Store (https://play.google.com/) - Malware samples found by [50], and the Contagio malware dump (http://contagiominidump.blogspot.com).
35	[9]	Malware signatures as the source dataset used in this reference.	Generate malware signatures using two free shareware products developed by the Honeynet Project, which are DroidBox (http://www.honeynet.org/gsoc/slot11), and Androguard (http://code.google.com/p/androguard/wiki/Usage)
36	[42]	- They used a publicly available non-official application programming interface (API), in addition to a set of PHP scripts in order to collect data - Obtained 9512 applications and related to 35 application categories containing between 190 and 590 applications each.	From the Google Play store (Android Market) in 2013 [55]
Datasets used within URL detection technique			
37	[43]	- 18 URL Features for Inspection were extracted from the collected data. These features include Hostname, Path Tokens, in addition to Primary Domain, TLD, URL Length and Network Features. Some of the features include Whois info (Registrar, Registrant, Registration Status, Whois Server, Registration Update Date, Registration Creation Date, Registration Expiration Date), IP Prefix, AS number, Geographic location, Communication link, Data Rate).	- this reference used A WebCrawler to gather a large amount of mobile specific URLs. - More than 2.4 million mobile specific unique URLs were gathered and stored in Hadoop Distributed File System in the university cloud data centre
38	[44]	- 11,361 phishing URLs were collected first set of ("OldPhishTank" data set). Phishing tactics used by scammers evolve over time; to track these evolving URL features - They collected second batch of 5,456 phishing URLs ("NewPhishTank" data set). - Non-phishing URLs were also collected - They use 22,213 legitimate URLs using (Yahoo data set). - They use 9,636 randomly chosen non-phishing URLs from DMOZ, a directory whose entries are vetted by editors (DMOZ data set)	- They coded scripts to automatically download confirmed phishing websites' URLs from PhishTank PhishTank (http://www.phishtank.com) - Non-phishing URLs were collected from Yahoo! directory and DMOZ Open Directory Project.
39	[45]	Datasets contain a total of 68,031 malicious URLs, and 122,550 benign URLs.	- They drew data from six sources. - Phishing data was collected from www.Phishtank.com, www.OpenPhish.org, www.MalwareDomainlist.com, and www.MalwareDomains.com. - Benign data set collected links from the www.Dmoz.org, and www.Alexa.com.
40	[46]	- Used data set with different labeling rate. - Including 138925 normal URL and 24520 malicious URL.	The experimental data set comes from a well-known Chinese Internet security company.

For different malware detection experiments, researchers have been used datasets with different trends. For benign data and applications, various benign applications categories are gathered from several public-source markets, the most common markets used are Google play and Anzhi. Malwares were collected from different malware repository; the most frequently used are Contagio and android malware Genome. On other hand, A harvested dataset from the literature was created by some researchers, public datasets were relied on by some of them, and the other ones relied on experiments to generate their own datasets. Generally, the most significant element in detecting malware app is the dataset, and that is due to capability of the author to prove the efficiency of their technique through these data. Lastly, public datasets are important and having them is a great opportunity for researchers to utilize them for validation and evaluation purposes for their methods. Some datasets were obtained from other studies such as, [47, 50, 51, 55] [49, 51-54]; the most frequently datasets used are provided by [50] and [51], respectively.

5. OPEN ISSUES

Lately, the benchmarking and evaluation field of malware detection applications over smartphone has been rapidly boosting, even though it still face issues and problems in various aspects yet. One of the major problems that has been faced the malware detection apps regarding to evaluation aspects is how to compare the current detection approach with the others in order to determine which are the detection approach is better [11], in the other words, how can benchmarking developed detection approach with the previous. Basically, this benchmarking process primarily depends on comparing between new generation and the others, in the same time it also has to consider the conditions and the criteria after the process for the development of any system [13]. In addition, the main challenges for malware detection applications development over smartphone are that the developers focused on either increase reliability of the application that has minimal rate of error or only reducing the time complexity [56]. This approach frequently poses effects of the results for the application of malware detection with high reliability and minimal rate (error rate or time complexity rate) that cannot be simultaneously achieved [7]. Therefore, this trade-off has reflection on the benchmarking process. Many studies face the issue of criteria conflict during the benchmarking which results in major challenges [7, 48, 56], in addition to the fact that due to the measurement of other criteria that generates set of numbers that also displays different criteria. Furthermore, there is a need to eliminate the cases among various criteria which in turn can affect the process of benchmarking. The most significant issues for evaluation criteria and benchmarking for the area of malware detection apps over smartphone are comprehensively explained in the following sub-sections.

5.1 Concern for Evaluation Criteria

When it comes to the metric of evaluation, the criteria used for the evaluation purpose of malware detection system over smartphone were robustly criticized. Various numbers of these criticisms were meant for evaluation criteria, in specific the error rate inside the metrics of the dataset. A figure issue exists for the vales of error rate variation within dataset which were results of dataset variation size utilized for malware detection experiments purposes. That is why lacking in the dataset standards could result in serious issues while on the other hand the value of error rate during various experiments are considered. Furthermore, datasets are gathered by researchers according to specific studies purposes, and that results in needless time and effort consumption. Because Some evaluation criteria including accuracy, precision, true positive rate, true negative rate, false positive rate, false negative rate...etc, relies on the parameters parts matrix like (TP, FP, TN, and FN) [8, 28] [21] [17] [38] [23]. and these four parameters are prone to lose values in experiments and it will lead to an effect over the outcomes by all the other criteria [32]. It's also computed according to the four parameters and this issue raises a debate. Despite the fact that there are large criticisms over the literature with respect to these parameters, these studies remain to use them for evaluation of malware detections apps and in other domains [8, 14].

5.2 Concern for Trade-off Criteria

The issue of "trade-off" is defined as a situation when a reliability or aspect of something decreased whiles the reliability or aspect of another increases. According to this literature review, it's found that different aspect of trade-off utilized by researchers for different criteria were performed, which in turn were confusing for decision makers. In addition, in our study the different use ratio in different criteria demonstrated effect that explains the conflict on other criteria utilized by researchers. Thus, the evaluation criteria conflict for malware detection over smartphone shows significance challenges in our intention towards creating a malware approach for detection. Fundamentally, these types of challenges are due to terms confliction, especially the one between the criteria not to mention the data. Thus, it's crucial to realize the advantages and disadvantages of particular choice while making a decision. The trade-off term is frequently used in the context of evaluation, where the process of selection acts as "decision maker [7, 48, 56]. The trade-off or as it's also called conflicting criteria problem between the evaluation criteria concentrated on the application reliability, complexity

of time for the malware detection application, in addition to the error rate within the dataset in the benchmarking and evaluation of application for malware detection over smartphone, are clearly reported in the mentioned studies [13, 57, 58]. With the aim of evaluation the malware detection applications, these sorts of criteria are considered main necessities [8, 14, 19]. The reliability should possess a high rate; time complexity to conduct the output that also need to below. In addition, the apparent error rate from the training of the dataset has to be simultaneously low. The generated Conflicting data is monitored because that matrix of parameters section contain TP, FP, TN, and FN, which displays the rise in TP and TN when parameter FP and FN are minimized [25, 58]. This phenomenon shows an apparent conflict amongst the probability criteria. These parameters have a considerable effect on some of the remaining criteria values since some of the criteria rely on the values of these four parameters. Therefore, the process of evaluation and benchmarking must take into considerations such requirements. Every study that was reviewed explained that all criterions evaluation and benchmarking depend on the general framework. For this reason, the malware detection mechanism over smartphone should be performed in order to standardize the basic and advanced requirements, in addition to clear methodology which in turn should be applied in the midst of research for phases including testing, evaluation, and benchmarking. As results, a new approach for evaluation that handles all conflict criteria and data problems should emerge and this method should be flexible. However, in this regard there are no suggested solutions to handle these particular issues.

5.3 Concern for Criterion Importance

While exploring the malware detection apps over smartphone studies, various objectives were considered during the planning phases. These objectives reflect within the system; design, evaluation and benchmarking. The key objective of this study is associated with the importance of the criteria through the evaluation and benchmarking phases despite their conflict. In addition, this conflict between the criteria poses a significant challenge during the evaluation stage [59]. There is a need for the development of a suitable procedure for this kind of objectives while boosting the significance of a certain evaluation criteria and minimizing the other one [60]. Two major key points must be considered. The first one is to achieve a sufficient understanding of the malware detection application behavior, while assigning certain significance to the design. The next point is the evaluation approach while bearing in mid the issue of trade-off. However, there might be a conflict between the opinions of the evaluator along with the objective of the designer in which poses an effect over the last evaluation of the needed approach [61]. From technical point of view, the detection application for malware by means of evaluation and benchmarking simultaneously considers multiple attributes and then assign a suitable weight for all the benchmarking approaches features for malware detection over smartphone. After making a comparison for all the approaches scores, the approaches with most balancing rate should be assigned with highest priority level, while on the other hand the approaches with the least balancing rate should be assigned with lowest level of priority. In addition, due to the fact that malware detection methods over smartphone have to consider multiple attributes, it considered as a difficult and challenging task in time and error rate in the dataset which also could be significantly important in the malware detection. In addition, each decision maker assigned a different weight for all these previous attributes [62]. On the other hand, the developers who in charge of assigning a score for the malware detection method cold assign more weights to different features aside from the ones that acquired less interest than any other attributes. By contrast, developers who aim to make use of software benchmarking in order to address such problems would consider different attributes as most significant ones. Thus, the process for evaluation and benchmarking for malware detection approach over smartphone could face a multi-complex attribute problem, like that all the approaches are considered to be an available alternative for the decision maker.

6. FUTURE RESEARH DIRECTION

This section is meant to describe the recommendation path solution of this research. The reviews to support are presented including the process of evaluation and benchmarking for the system of malware detection over smartphone that involves simultaneous consideration for multi criteria ("reliability, time complexity rate, and error rate within dataset") with the aim of evaluating and scoring systems of malware detection over smartphone. Therefore, adapting candid and structured techniques for decisions with the use of multiple attributes which could boost the decision-making quality, in addition to set of methods identified under the collective heading multi criteria decision analysis (MCDA), are usable in situations like this. This, the appropriate methods which address issues of CDM are shown as recommended solutions and pathways that collectively aid the decision makers in order to organize any problem and have it solved, in addition to applying analysis, assessment and ranking [63].

6.1 Definition and Significant of Multi-Criteria Decision Making

Both Keeney and Raiffa [61] as "an extension of decision theory that covers any decision with multiple objectives. A methodology for assessing alternatives on individual, often conflicting criteria, and combining them into one overall appraisal…" Furthermore, in Belton and Stewart [64] MCDM is defined as "an umbrella term to describe a collection of formal approaches, which seek to take explicit account of multiple criteria in helping individuals or groups explore decisions that matter."

The techniques of decision making are widely recognized and among the most significant ones is the MCDM, it's also considered as important part of operation research that handles problems of decision making with respect to decision criteria [65, 66], the techniques is involved in various processes including structuring, planning, in addition to solving different decision problems with the sue of many attributes [66]. There is a considerable global rise in the use of multi criteria decision making since it's able to promote the decision quality. Its achieved by making the process of the decision more reasonable, efficient, clear and explicit in compare with other traditional processes [67]. The most significant goals of multi criteria decision making includes the assigning of the data minter in order to choose the most suitable alternatives, in addition to assigning a rank to the alternatives in decreasing order in regards to the efficiency, and classifying the applicable alternatives amongst groups of the available alternatives.[63] [63] [63]. Based on that, the ranking will take place on the most suitable alternative(s).there is a need for the fundamental terms in MCDM to be defined, in addition to containing the decision matrix, and its associated attributes [68].

There is an improvement possibility for the decision-making process by means of comprising both decision makers and stakeholders which will enable the process with support and structure. With the use of Candid, the structure of the multi criteria decisions methods can aid towards improving the decision-making quality and set of techniques. These techniques could provide clearly in identifying which of the criteria are relevant, in addition to the significance of each, and how a framework can involve this information for purposes of evaluating the current alternatives. By doing this, they are able to aid the transparency increase, in addition to consistency and decision validity. MCDM can contribute to processes which are fair, transparent and rational priority-setting processes. MCDM has been recognized for its common utilization in many areas for different applications.

6.2 MCDM Methods

Different theories of MCDM are discovered. The following figure 4 illustrates the mostly used and famous methods of MCDM that use different concepts:

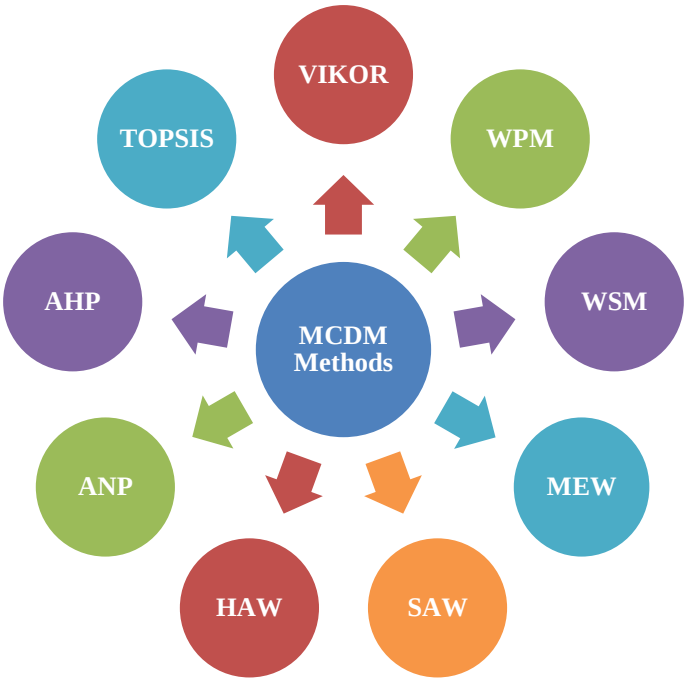

Fig.4 The most popular and famous MCDM methods

The MCDM methods advantages and limitations are presented as follows in Table 3 according to the previous studies [69].

Table 3 Limitations and advantages of MCDM techniques

MCDM METHODS		ADVANTAGES	DISADVANTAGES
	HAW & WSM	• Easy To Understood and Use	• Arbitrarily assigned attribute weights • Difficult adoption with numerous criteria • Use of common numerical scaling in calculating final score
	WPM & MEW	• Capability of eliminating any element to be measured • Use of proportional (rather than real or actual) values	• Incapability of providing any solution with equal DM weight
	SAW	• Consideration of all criteria/attribute • Simple calculation • Intuitive decision making	• Need for positive and maximum values for all criteria • Common incapability of discovering real situation
	AHP	• Empower the Decision Making in order enable the structuring of decision-making problems into hierarchy trees • Facilitation of understanding of problems • Time-consuming support caused by large number of pairwise comparisons and need for mathematical calculations, which increase with number of attributes or alternatives • Substantial restriction imposed by human capacity for information processing (7+/- 2 is regarded as comparison ceiling)	• Dependency of scoring and ranking on alternatives considered for evaluation • Potential change in final ranking caused by removal or addition of alternatives (rank reversal problem) •
	ANP	• Provision of full understanding of importance level that can be assumed by an attribute with respect to its correlation with other attributes	• Offering proper network structure has its Complexity between attributes even for experts (Different structures lead to varying results.)

			• Enabling the measurement of judgments' consistency. The measurements of consistency cannot be evaluated when weights are specified by compromise • Specifying weights Assistance by separating the problem into small parts such that experts can have manageable discussion because only two attributes are compared in specifying judgments		• Formation of super matrix's Need to be pairwise comparison for all attributes with all other attributes (complex and unnatural process)
	TOPSIS &VIKOR		• Significant approaches to solving real-world problems • Application in discretizing alternative challenges • Capability of immediately recognizing proper alternative • Decrease in number of required pairwise comparisons, with capacity limitation not necessarily controlling the process • Useful when alternatives and attributes are numerous and when quantitative or objective data are available • Basis in aggregating function representing 'closeness to the ideal', which originates from compromise programming method		• TOPSIS and VIKOR include the lack of provision to weigh elicitation and check the consistency of judgments • TOPSIS does not consider the relative importance of distances.

According to the performed analysis we conducted, all the presented methods in the literature did not used for purposes of evaluation and benchmarking for malware techniques of detection over smartphone. The techniques are based on machine learning techniques. These Methods are challenged by non-adoption requirement-driven approach which make them to be unsuitable for measurement and scoring in decision making [69]. However, for cases that involve numerous alternatives and criteria, both TOPSIS and VIKOR are applicable. It's convenient to utilize both methods, VIKOR and TOPSIS when the given data are quantitative or objective. TOPSIS is able to create a shortest distance solution towards the ideal solution and also the largest distance away from the negative-ideal solution. Nevertheless, there is no consideration for the relative significance of these distances [70]. On the other hand, the other technique, VIKOR has functional relationship to the discrete-alternative problems. For VIKOR, it's the most practical routes techniques that works and operate with the aim of solving real world problems. The advantage of VIKOR is that it's able to rapidly decide the best alternative. Furthermore, VIKOR is suitable technique for cases where many alternatives and attributes situations [70]. Nevertheless, the major drawback of VIKOR is its lack in provisioning for elicitation of weight and checking for judgment consistency [70]. Thus, VIKOR needs an effective technique in order to acquire the relative importance for various criteria with respect to the objective, and AHP is able to provide such a technique. However, AHP is utilized for setting objectives weights on the preferences basis of the stakeholder [71], and it is restricted majorly by the human capacity for information processing; therefore the 7 ± 2 would be the comparison ceiling [72]. In All the optimization problems of multi criteria, decision maker implicitly aims towards identifying a solution of the given criteria which can be satisfactory up to the most possible extent. The decision maker also has to make sure that he doesn't violate the existing limitations. These problems sadly has no unified global solution i.e., and optimal solution doesn't exist for all the criteria at the same time. It occasionally occurs because of the differences of nature for some of the criteria that are differently expressed in units of measurement, from monetary units through physical size units, to probability or subjective evaluations determined on the basis of a scale formed for a specific problem[73]. The presence of several criteria may itself negatively impact the rational comparison of alternatives by a DM. This possible confusion or uncertainty may lead to a naïve approach of simply adding up pluses and minuses. Whilst estimating preferences, such uncertainty may also introduce cognitive dissonance i.e. the holding of two contradictory beliefs simultaneously [73-78]. To overcome these issues, it necessary to apply fuzzy set to overcome the vagueness in the decision-making practice because the vagueness and imperfect information in decision making. As well as, many authors confirmed that the fuzzy set theory enables to handle imprecision of evaluations and avoid or reduce the uncertainties and ambiguities surrounding these kinds in decision-making process [73, 79-82]. According to above, this study attempts to use the fuzzy theory with AHP, which represents the most common used weighting method in order to deal with uncertainty issue that faced the experts when they try to making the preference comparison amongst the criteria. Recently, there have been new trends in the use of MCDM approaches which are capable of to integrate two or more

methods that are able to compensate the single method shortcomings [69]. The use of Fuzzy AHP-VIKOR became well known integrated method of MCDM used for various reasons, such as using data of weights and objectives orderly with the aim of acquiring relative distance, its ability to provide full Ranking results, it's the trade-off smoothing by addressing nonlinear relationships [83-85]. In addition, there are many integration approaches for Fuzzy AHP-VIKOR which are involved in measuring the alternative and ranking cases in the previous studies [83-85]. Lastly, in order to evaluate and benchmark system used for malware detection over smartphone, its recommended to integrate AHP method for assigning weights and then have them distributed for evaluation attributes/criteria purposes including ("reliability, time complexity rate, and error rate within dataset") by relying of the judgment of an expert, and VIKOR method that is needed to offer a comprehensive ranking of malware detection systems.

7. METHODOLOGY

7.1 Conceptual framework

This section describes and explains the methodological of the evaluation and selection methodology of malware detection techniques over smartphone. The output ranked malware detection techniques based on our set of criteria using the AHP ranking. Figure 4 illustrates all the elements of our study in the overall conceptual. According to our conceptual framework in figure 5, two steps will be performed to develop our methodology for evaluation and selection the malware detection techniques over smartphone, firstly, construct decision matrix, secondly, using AHP to calculate the weights for alternatives and VIKOR to rank the malware detection techniques.

7.1.1 Construct Decision Matrix

Decision matrix considers the main component in our methodology of evaluation and selection of malware detection techniques over smartphone. The components of the decision matrix are comprised on components which are decision alternatives and decision criteria. In our case, the malware detection techniques are the decision alternatives, and the criteria are evaluation criteria identified based on previous studies. The alternatives in decision matrix will be malware detection techniques. The evaluation criteria are identified based in our deep analysis for related previous publications that will be used in our decision matrix namely, accuracy, error rate, precision, FP, FN, TP, TN, F score , False Positive Rate, False Negative Rate, True Positive Rate , False Positive Rate , Training Time, detection Time, Area Under Curve. In order to construct the decision matrix, 13 malware detection techniques will be developed based on most frequent approaches of machine learning which can utilize malware detections as alternatives for decision in our decision matrix. In General, there are two-step processes towards the development of malware detections techniques. The first process is the recognition for the process of training (learning). The malware detection techniques which describe a predetermined class set are built, it is done through analyzing the training instances for the dataset. Each individual instance must belong to a predefined class. Every instance is expected to belong to class which is predefined. In the second process, the techniques for malware detection over smartphone are running with the use of other dataset that is independent recognized as dataset testing in order towards perfuming malware detection technique estimation. If the detection technique of the malware performance looks 'acceptable', the malware technique for detection, the technique can be used in classifying future data for which the class label is unknown. Ultimately, the technique of malware detection over smartphone in which it can provide results which are acceptable can be considered an acceptable malware detection technique. Regarding to the dataset that will be used to apply the malware detection techniques which will be developed, Choosing the data set will be depend on which type of the malware detection techniques will be developed, and table 1 in section 4 illustrated various dataset used in different reviewed studies, this study can follow any of them. 13 malware detection techniques will be built while relying on well-known machine learning methods. These machine learning methods have been used extensively in previous literature, and all these methods have displayed good results when they are used in the malware dataset classification; they include Bayesian Network, Random Forests, Support Vector machine, Logistic Model, K-Nearest Neighbour, Naïve Bayes, Logistic regression, artificial neural network, Random Tree, Simple Logistic, Gaussian mixture model, decision tree, Decision stump. All The following details are concerned with each method. In order to develop malware detection technique, the dataset will be separated for two parts. The first part will be used towards training the set. The second part will be used for testing for the set. The first part where the set is being trained will be used in the techniques of malware detection. The second part of the dataset that was the testing set will be used in order to test the malware detection techniques which were trained previously. The 13 built malware detection technique will classify the test dataset into two categories. The first one is malware and the second one is non- malware. In order to construct the decision matrix, a crossover is performed among various identified evaluation criteria with various developed malware detection technique. The major components in the decision matrix which will be constructed are the alternatives and criteria. The alternatives

now are thirteen malware detection techniques, and the criteria are fifteen of the criteria. Figure 5 presents the structure of the decision matrix.

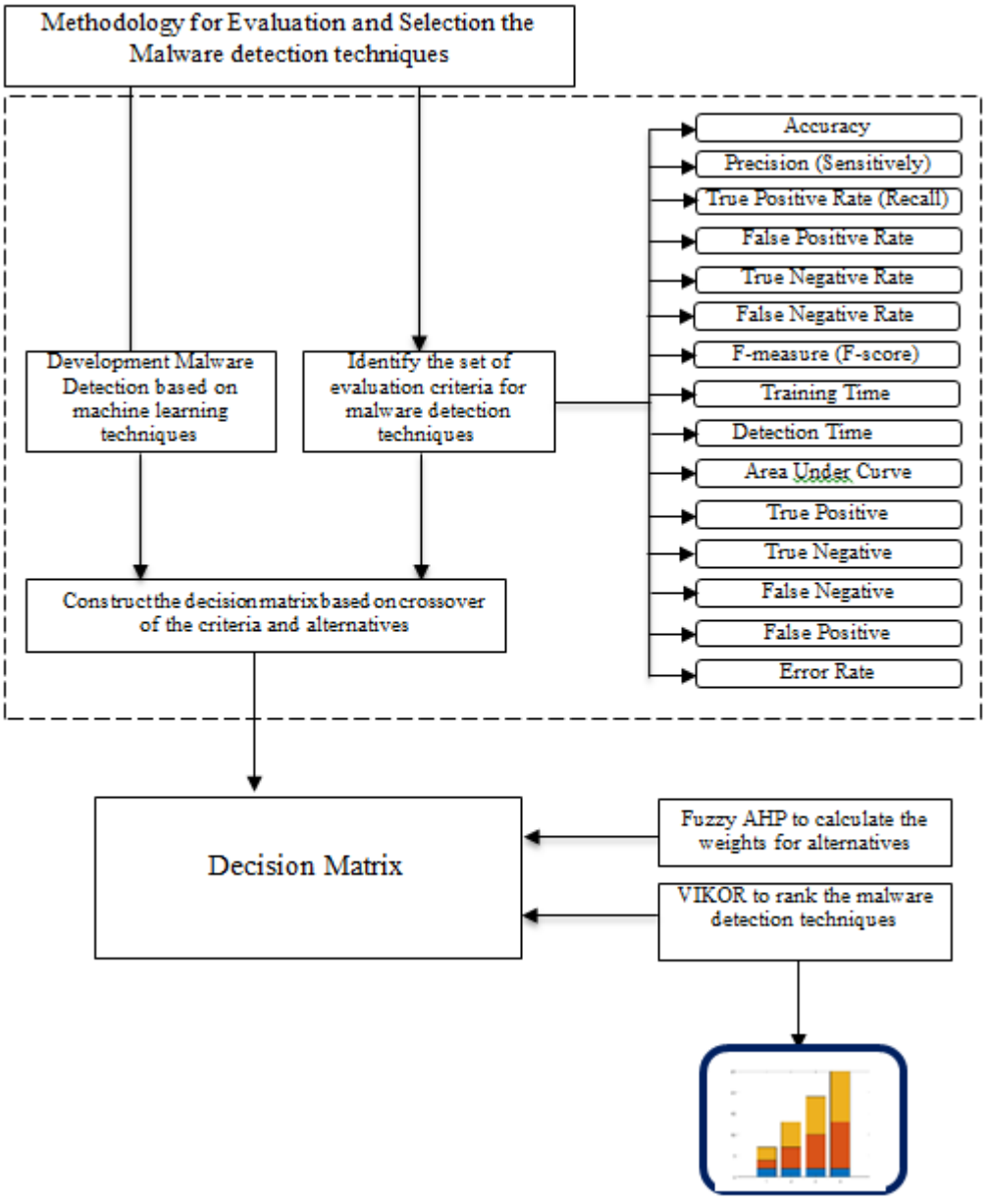

Fig.4 Conceptual framework

Criteria / Alternatives	ER	precision	FP	FN	TP	TN	Fscore	FPR	FNR	TPR	FPR	TT	DT	AUC
Bayesian Network														
Random Forest														
Support Vector machine														
Logistic Model														
K-Nearest Neighbour														
Naive Bayes														
Logistic regression														
artificial neural network														
Random Tree														
Simple Logistic														
Gaussian mixture model														
decision tree														
Decision stump														

ER: error rate, precision, FP, FN, TP, TN, fscore , FPR: False Positive Rate, FNR: False Negative Rate, TPR: True Positive Rate , FPR: False Positive Rate , TT: Training Time, DT: Detection Time, AUC: Area Under Curve.

Fig.5 proposed the decision matrix

Figure 6 illustrates the proposed decision matrix; the values inside the decision matrix will be the results of evaluation the performance of those 13 malware detection techniques.

7.1.2 Integrated Fuzzy AHP and VIKOR

To develop the methodology for evaluation and selection the malware detection techniques using multi-criteria decision-making analysis. Thus, our methodology will be developed based on integration of AHP and VIKOR techniques for ranking and selecting the best alternatives in the proposed decision matrix. Figure 6 illustrates the methodology for evaluation and selection the malware detection techniques.

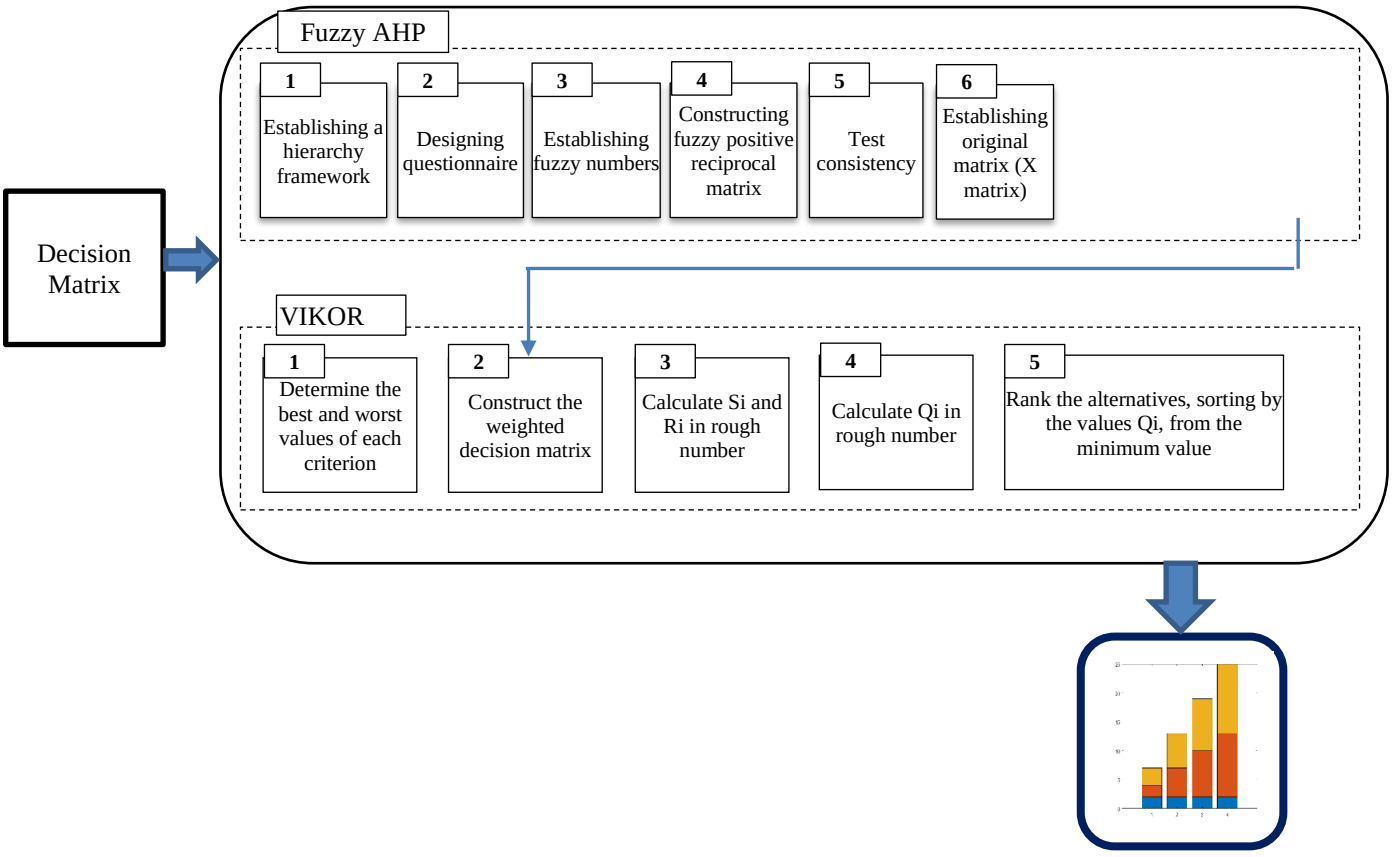

Fig.6 Framework of evaluation and selection the malware detection techniques over smartphone

8. CONCLUSION

The mains aim of this research includes the surveying of the efforts of different researchers with respect to new and emerging malware detection techniques from machine learning point of view over smartphone environment, in addition to the terms used in evaluating and benchmarking. The aim was to map the landscape of research acquired from the literature, and then construct a coherent taxonomy. This research offers a taxonomy that aims to shade lights and review multiple criteria for the evaluating and benchmarking of malware detection approaches over smartphone, the taxonomy is structured into three layers including type of detection technique, various machine learning used in detection techniques over smartphones, and multiple criteria utilized for evaluation malware detection techniques. In addition, this research presented a statistical number regarding the distribution of different criteria while considering identified classes. Furthermore, the findings of this research confirmed three of the main open issues associated with evaluation criteria and benchmarking for malware detection approaches, which in turn are explained including evaluation concerns of criteria, trade-off criteria concerns, and criteria importance concerns. In addition, MCDM in the framework of malware detection techniques over smartphone with respect to evaluation and benchmarking were discussed. Many techniques in the area of decision making exhibited different contexts and configurations. Therefore, the appropriate useful approaches addressing issues related to MCDM, which were shown as suitable suggested pathways and solutions to aid the decision maker towards organizing any issue and have it solved and also apply analysis, ranking and assessment. The proposed methodology is recommended as a new solution, includes integrating of fuzzy AHP and Group-VIKOR in order to assign and distribute attributes/criteria evaluation weights based on the judgement of experts, and also to make use of VIKOR method which requires a comprehensive ranking for the malware detection techniques over smartphone.

Funding

The author had no institutional or sponsor backing.

Conflicts of Interest

The author's disclosure statement confirms the absence of any conflicts of interest.

Acknowledgment

The author extends appreciation to the institution for their unwavering support and encouragement during the course of this research.

References

[1] S. Wu, P. Wang, X. Li, and Y. Zhang, "Effective detection of android malware based on the usage of data flow APIs and machine learning," Information and Software Technology, vol. 75, pp. 17-25, 2016/07/01/ 2016.
[2] K. Sharma and B. Gupta, "Multi-layer Defense Against Malware Attacks on Smartphone Wi-Fi Access Channel," Procedia Computer Science, vol. 78, pp. 19-25, 2016.
[3] P. Farina, E. Cambiaso, G. Papaleo, and M. Aiello, "Are mobile botnets a possible threat? The case of SlowBot Net," Computers & Security, vol. 58, pp. 268-283, 5// 2016.
[4] Q. Do, B. Martini, and K.-K. R. Choo, "Exfiltrating data from Android devices," Computers & Security, vol. 48, pp. 74-91, 2// 2015.
[5] S.-H. Hung, S.-W. Hsiao, Y.-C. Teng, and R. Chien, "Real-time and intelligent private data protection for the Android platform," Pervasive and Mobile Computing, vol. 24, pp. 231-242, 2015.
[6] S. Ming-Yang and C. Wen-Chuan, "Permission-based malware detection mechanisms for smart phones," in The International Conference on Information Networking 2014 (ICOIN2014), 2014, pp. 449-452.
[7] S. Y. Yerima, S. Sezer, and I. Muttik, "High accuracy android malware detection using ensemble learning," IET Information Security, vol. 9, pp. 313-320, 2015.
[8] H.-S. Ham, H.-H. Kim, M.-S. Kim, and M.-J. Choi, "Linear SVM-based android malware detection for reliable IoT services," Journal of Applied Mathematics, vol. 2014, 2014.
[9] P. Wang and Y.-S. Wang, "Malware behavioural detection and vaccine development by using a support vector model classifier," Journal of Computer and System Sciences, vol. 81, pp. 1012-1026, 2015/09/01/ 2015.
[10] D. Hang, N.-q. HE, H. Ge, L. Qi, and M. ZHANG, "Malware detection method of android application based on simplification instructions," The Journal of China Universities of Posts and Telecommunications, vol. 21, pp. 94-100, 2014.
[11] K. A. Talha, D. I. Alper, and C. Aydin, "APK Auditor: Permission-based Android malware detection system," Digital Investigation, vol. 13, pp. 1-14, 2015.
[12] M. Abuthawabeh and K. W. Mahmoud, "Enhanced android malware detection and family classification, using conversation-level network traffic features," Int. Arab J. Inf. Technol., vol. 17, pp. 607-614, 2020.
[13] C. Wang, Z. Li, X. Mo, H. Yang, and Y. Zhao, "An android malware dynamic detection method based on service call co-occurrence matrices," Annals of Telecommunications, vol. 72, pp. 607-615, 2017.
[14] B. Shrestha, M. Mohamed, A. Borg, N. Saxena, and S. Tamrakar, "Curbing mobile malware based on user-transparent hand movements," in Pervasive Computing and Communications (PerCom), 2015 IEEE International Conference on, 2015, pp. 221-229.
[15] T. E. Wei, C. H. Mao, A. B. Jeng, H. M. Lee, H. T. Wang, and D. J. Wu, "Android Malware Detection via a Latent Network Behavior Analysis," in 2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications, 2012, pp. 1251-1258.
[16] A. E. Attar, R. Khatoun, and M. Lemercier, "Trimming Approach of Robust Clustering for Smartphone Behavioral Analysis," in Embedded and Ubiquitous Computing (EUC), 2014 12th IEEE International Conference on, 2014, pp. 315-320.
[17] G. Ajaeiya, I. H. Elhajj, A. Chehab, A. Kayssi, and M. Kneppers, "Mobile Apps identification based on network flows," Knowledge and Information Systems, pp. 1-26, 2018.
[18] H. Papadopoulos, N. Georgiou, C. Eliades, and A. Konstantinidis, "Android malware detection with unbiased confidence guarantees," Neurocomputing, vol. 280, pp. 3-12, 2018.
[19] A. Shastry, M. Kantarcioglu, Y. Zhou, and B. Thuraisingham, "Randomizing smartphone malware profiles against statistical mining techniques," in IFIP Annual Conference on Data and Applications Security and Privacy, 2012, pp. 239-254.
[20] M. S. Alam and S. T. Vuong, "Random Forest Classification for Detecting Android Malware," in Green Computing and Communications (GreenCom), 2013 IEEE and Internet of Things (iThings/CPSCom), IEEE International Conference on and IEEE Cyber, Physical and Social Computing, 2013, pp. 663-669.

[21] Y. Lu, P. Zulie, L. Jingju, and S. Yi, "Android Malware Detection Technology Based on Improved Bayesian Classification," in Instrumentation, Measurement, Computer, Communication and Control (IMCCC), 2013 Third International Conference on, 2013, pp. 1338-1341.
[22] H. Kurniawan, Y. Rosmansyah, and B. Dabarsyah, "Android anomaly detection system using machine learning classification," in Electrical Engineering and Informatics (ICEEI), 2015 International Conference on, 2015, pp. 288-293.
[23] H. Wang, H. He, and W. Zhang, "Demadroid: Object Reference Graph-Based Malware Detection in Android," Security and Communication Networks, vol. 2018, 2018.
[24] F. Yuan, L. Zhai, Y. Cao, and L. Guo, "Research of Intrusion Detection System on Android," in 2013 IEEE Ninth World Congress on Services, 2013, pp. 312-316.
[25] Y. Wei, Z. Hanlin, G. Linqiang, and R. Hardy, "On behavior-based detection of malware on Android platform," in 2013 IEEE Global Communications Conference (GLOBECOM), 2013, pp. 814-819.
[26] A. Shabtai, L. Tenenboim-Chekina, D. Mimran, L. Rokach, B. Shapira, and Y. Elovici, "Mobile malware detection through analysis of deviations in application network behavior," Computers & Security, vol. 43, pp. 1-18, 2014/06/01/ 2014.
[27] A. E. Attar, R. Khatoun, B. Birregah, and M. Lemercier, "Robust clustering methods for detecting smartphone's abnormal behavior," in 2014 IEEE Wireless Communications and Networking Conference (WCNC), 2014, pp. 2552-2557.
[28] G. Nguyen, B. M. Nguyen, D. Tran, and L. Hluchy, "A heuristics approach to mine behavioural data logs in mobile malware detection system," Data & Knowledge Engineering, vol. 115, pp. 129-151, 2018.
[29] M. Fan, J. Liu, W. Wang, H. Li, Z. Tian, and T. Liu, "DAPASA: Detecting Android Piggybacked Apps Through Sensitive Subgraph Analysis," IEEE Transactions on Information Forensics and Security, vol. 12, pp. 1772-1785, 2017.
[30] D. Damopoulos, G. Kambourakis, S. Gritzalis, and S. O. Park, "Exposing mobile malware from the inside (or what is your mobile app really doing?)," Peer-to-Peer Networking and Applications, vol. 7, pp. 687-697, December 01 2014.
[31] S. Y. Yerima, S. Sezer, G. McWilliams, and I. Muttik, "A new android malware detection approach using bayesian classification," in Advanced Information Networking and Applications (AINA), 2013 IEEE 27th International Conference on, 2013, pp. 121-128.
[32] M. Fazeen and R. Dantu, "Another free app: Does it have the right intentions?," in Privacy, Security and Trust (PST), 2014 Twelfth Annual International Conference on, 2014, pp. 282-289.
[33] N. Peiravian and X. Zhu, "Machine Learning for Android Malware Detection Using Permission and API Calls," in 2013 IEEE 25th International Conference on Tools with Artificial Intelligence, 2013, pp. 300-305.
[34] S. Sheen, R. Anitha, and V. Natarajan, "Android based malware detection using a multifeature collaborative decision fusion approach," Neurocomputing, vol. 151, pp. 905-912, 2015/03/05/ 2015.
[35] L. Wen, "Mutiple classifier system based android malware detection," in 2013 International Conference on Machine Learning and Cybernetics, 2013, pp. 57-62.
[36] W. Wang, Y. Li, X. Wang, J. Liu, and X. Zhang, "Detecting Android malicious apps and categorizing benign apps with ensemble of classifiers," Future Generation Computer Systems, vol. 78, pp. 987-994, 2018.
[37] S. Chen, M. Xue, L. Fan, S. Hao, L. Xu, H. Zhu, et al., "Automated poisoning attacks and defenses in malware detection systems: An adversarial machine learning approach," computers & security, vol. 73, pp. 326-344, 2018.
[38] G. Kirubavathi and R. Anitha, "Structural analysis and detection of android botnets using machine learning techniques," International Journal of Information Security, vol. 17, pp. 153-167, 2018.
[39] X. Su, M. Chuah, and G. Tan, "Smartphone Dual Defense Protection Framework: Detecting Malicious Applications in Android Markets," in Mobile Ad-hoc and Sensor Networks (MSN), 2012 Eighth International Conference on, 2012, pp. 153-160.
[40] X. Wang, Y. Yang, and Y. Zeng, "Accurate mobile malware detection and classification in the cloud," SpringerPlus, vol. 4, p. 1, 2015.
[41] M. Lindorfer, M. Neugschwandtner, and C. Platzer, "MARVIN: Efficient and Comprehensive Mobile App Classification through Static and Dynamic Analysis," in Computer Software and Applications Conference (COMPSAC), 2015 IEEE 39th Annual, 2015, pp. 422-433.
[42] K. Sokolova, C. Perez, and M. Lemercier, "Android application classification and anomaly detection with graph-based permission patterns," Decision Support Systems, vol. 93, pp. 62-76, 2017.

[43] H. Adas, S. Shetty, and W. Tayib, "Scalable detection of web malware on smartphones," in Information and Communication Technology Research (ICTRC), 2015 International Conference on, 2015, pp. 198-201.
[44] R. B. Basnet and T. Doleck, "Towards developing a tool to detect phishing URLs: a machine learning approach," in 2015 IEEE International Conference on Computational Intelligence & Communication Technology, 2015, pp. 220-223.
[45] M. Darling, G. Heileman, G. Gressel, A. Ashok, and P. Poornachandran, "A lexical approach for classifying malicious URLs," in High Performance Computing & Simulation (HPCS), 2015 International Conference on, 2015, pp. 195-202.
[46] J. Yang, P. Yang, X. Jin, and Q. Ma, "Multi-classification for malicious URL based on improved semi-supervised algorithm," in Computational Science and Engineering (CSE) and Embedded and Ubiquitous Computing (EUC), 2017 IEEE International Conference on, 2017, pp. 143-150.
[47] I. Burguera, U. Zurutuza, and S. Nadjm-Tehrani, "Crowdroid: behavior-based malware detection system for android," in Proceedings of the 1st ACM workshop on Security and privacy in smartphones and mobile devices, 2011, pp. 15-26.
[48] A. Shastry, M. Kantarcioglu, Y. Zhou, and B. Thuraisingham, "Randomizing Smartphone Malware Profiles against Statistical Mining Techniques," Berlin, Heidelberg, 2012, pp. 239-254.
[49] A. P. Felt, M. Finifter, E. Chin, S. Hanna, and D. Wagner, "A survey of mobile malware in the wild," in Proceedings of the 1st ACM workshop on Security and privacy in smartphones and mobile devices, 2011, pp. 3-14.
[50] X. Jiang and Y. Zhou, "Dissecting android malware: Characterization and evolution," in 2012 IEEE Symposium on Security and Privacy, 2012, pp. 95-109.
[51] D. Arp, M. Spreitzenbarth, M. Hubner, H. Gascon, K. Rieck, and C. Siemens, "DREBIN: Effective and Explainable Detection of Android Malware in Your Pocket," in Ndss, 2014, pp. 23-26.
[52] M. Zheng, M. Sun, and J. Lui, "Droidanalytics: a signature based analytic system to collect, extract, analyze and associate android malware," arXiv preprint arXiv:1302.7212, 2013.
[53] A. F. A. Kadir, N. Stakhanova, and A. A. Ghorbani, "Android botnets: What urls are telling us," in International Conference on Network and System Security, 2015, pp. 78-91.
[54] W. Wang, X. Wang, D. Feng, J. Liu, Z. Han, and X. Zhang, "Exploring permission-induced risk in android applications for malicious application detection," IEEE Transactions on Information Forensics and Security, vol. 9, pp. 1869-1882, 2014.
[55] android-market-api-php.
[56] M. Lindorfer, M. Neugschwandtner, and C. Platzer, "MARVIN: Efficient and Comprehensive Mobile App Classification through Static and Dynamic Analysis," in 2015 IEEE 39th Annual Computer Software and Applications Conference, 2015, pp. 422-433.
[57] H.-S. Ham, H.-H. Kim, M.-S. Kim, and M.-J. Choi, "Linear SVM-Based Android Malware Detection for Reliable IoT Services," J. Appl. Math., vol. 2014, Special Issue, 2014 2014.
[58] H. Kurniawan, Y. Rosmansyah, and B. Dabarsyah, "Android anomaly detection system using machine learning classification," in 2015 International Conference on Electrical Engineering and Informatics (ICEEI), 2015, pp. 288-293.
[59] M. Ilangkumaran, V. Sasirekha, L. Anojkumar, and M. Boopathi Raja, "Machine tool selection using AHP and VIKOR methodologies under fuzzy environment," International Journal of Modelling in Operations Management, vol. 2, pp. 409-436, 2012.
[60] H. E. Aktan and P. K. Samut, "Agricultural performance evaluation by integrating fuzzy AHP and VIKOR methods," International Journal of Applied Decision Sciences, vol. 6, pp. 324-344, 2013.
[61] R. L. Keeney and H. Raiffa, Decisions with multiple objectives: preferences and value trade-offs: Cambridge university press, 1993.
[62] M. Oliveira, D. B. Fontes, and T. Pereira, "Multicriteria decision making: a case study in the automobile industry," 2013.
[63] A. Jadhav and R. Sonar, "Analytic hierarchy process (AHP), weighted scoring method (WSM), and hybrid knowledge based system (HKBS) for software selection: a comparative study," in 2009 Second International Conference on Emerging Trends in Engineering & Technology, 2009, pp. 991-997.
[64] V. Belton and T. Stewart, "Multiple criteria decision analysis: an integrated approachKluwer Academic Publishers," ed: Boston, 2002.
[65] S. Petrovic-Lazarevic and A. Abraham, "Hybrid fuzzy-linear programming approach for multi criteria decision making problems," arXiv preprint cs/0405019, 2004.
[66] J. Malczewski, GIS and multicriteria decision analysis: John Wiley & Sons, 1999.

[67] S. Zionts, "MCDM-If not a Roman Numeral, then what?," Interfaces, vol. 9, pp. 94-101, 1979.
[68] M. Whaiduzzaman, A. Gani, N. B. Anuar, M. Shiraz, M. N. Haque, and I. T. Haque, "Cloud service selection using multicriteria decision analysis," The Scientific World Journal, vol. 2014, 2014.
[69] M. Aruldoss, T. M. Lakshmi, and V. P. Venkatesan, "A survey on multi criteria decision making methods and its applications," American Journal of Information Systems, vol. 1, pp. 31-43, 2013.
[70] S. Opricovic and G.-H. Tzeng, "Compromise solution by MCDM methods: A comparative analysis of VIKOR and TOPSIS," European journal of operational research, vol. 156, pp. 445-455, 2004.
[71] H. Nilsson, E.-M. Nordström, and K. Öhman, "Decision support for participatory forest planning using AHP and TOPSIS," Forests, vol. 7, p. 100, 2016.
[72] T. L. Saaty and M. S. Ozdemir, "Why the magic number seven plus or minus two," Mathematical and computer modelling, vol. 38, pp. 233-244, 2003.
[73] D. Pamučar, I. Petrović, and G. Ćirović, "Modification of the Best–Worst and MABAC methods: A novel approach based on interval-valued fuzzy-rough numbers," Expert systems with applications, vol. 91, pp. 89-106, 2018.
[74] N. Jaini and S. Utyuzhnikov, Trade-off ranking method for multi-criteria decision analysis vol. 24, 2017.
[75] M. Bashiri and H. Badri, "A group decision making procedure for fuzzy interactive linear assignment programming," Expert Systems with Applications, vol. 38, pp. 5561-5568, 2011.
[76] S. Guo and H. Zhao, "Fuzzy best-worst multi-criteria decision-making method and its applications," Knowledge-Based Systems, vol. 121, pp. 23-31, 2017/04/01/ 2017.
[77] Q. Yang, Z. Zhang, X. You, and T. Chen, "Evaluation and Classification of Overseas Talents in China Based on the BWM for Intuitionistic Relations," Symmetry, vol. 8, p. 137, 2016.
[78] Q. Mou, Z. Xu, and H. Liao, "An intuitionistic fuzzy multiplicative best-worst method for multi-criteria group decision making," Information Sciences, vol. 374, pp. 224-239, 2016.
[79] A. Hafezalkotob and A. Hafezalkotob, "A novel approach for combination of individual and group decisions based on fuzzy best-worst method," Applied Soft Computing, vol. 59, pp. 316-325, 2017.
[80] S. Ghaffari, A. Arab, J. Nafari, and M. Manteghi, "Investigation and evaluation of key success factors in technological innovation development based on BWM," Decision Science Letters, vol. 6, pp. 295-306, 2017.
[81] A. Sotoudeh-Anvari, S. Sadjadi, S. Molana, and S. Sadi-Nezhad, "A new MCDM-based approach using BWM and SAW for optimal search model," Decision Science Letters, vol. 7, pp. 395-404, 2018.
[82] H. Gupta, "Evaluating service quality of airline industry using hybrid best worst method and VIKOR," Journal of Air Transport Management, vol. 68, pp. 35-47, 2018.
[83] H.-P. Fu, K.-K. Chu, P. Chao, H.-H. Lee, and Y.-C. Liao, "Using fuzzy AHP and VIKOR for benchmarking analysis in the hotel industry," The Service Industries Journal, vol. 31, pp. 2373-2389, 2011.
[84] M. M. Fouladgar, A. Yazdani-Chamzini, E. K. Zavadskas, S. H. Yakhchali, and M. H. Ghasempourabadi, "PROJECT PORTFOLIO SELECTION USING FUZZY AHP AND VIKOR TECHNIQUES," Transformations in Business & Economics, vol. 11, 2012.
[85] K. Rezaie, S. S. Ramiyani, S. Nazari-Shirkouhi, and A. Badizadeh, "Evaluating performance of Iranian cement firms using an integrated fuzzy AHP–VIKOR method," Applied Mathematical Modelling, vol. 38, pp. 5033-5046, 2014.