WEIGHTED ENSEMBLE BASED EXTRA TREE FOR PERMISSION ANALYSIS FOR ANDROID APPLICATIONS CLASSIFICATION

[1]HOWIDA ABUBAKER, [2]AIDA ALI, [3]SITI MARIYAM SHAMSUDDIN

[1]Faculty of Engineering, School of Computing, Universiti Teknologi Malaysia (UTM), Johor Malaysia

[2]Faculty of Engineering, School of Computing, Universiti Teknologi Malaysia (UTM), Johor Malaysia

[3]Faculty of Engineering, School of Computing, Universiti Teknologi Malaysia (UTM), Johor Malaysia

E-mail: [1] howida10@gmail.com, [2]aida@utm.com, [3]mariyam@utm.my

ABSTRACT

Selecting optimal features for classification task is one of the essential problems in machine learning field. Feature Selection is one of the most extensively studied methods for dimensionality reduction. The feature selection method preserves a subset of the existing features and discards the rest during the (supervised or unsupervised) learning process. However, representing features plays important role in obtaining the highly discriminant features that contribute in enhancing the classifier performance. Therefore, the aim of this paper is to propose a framework based on ensemble extra tree algorithm to assign weight to features that have high influence in classifying android apps to malware or non-malware with lower computational cost overhead. The presented framework is evaluated by using different machine learning classifiers to examine the permissions features of two datasets in terms of their representation as binary vector or weight vector in enhancing the classification performance. The experimental results show that the presented model based on features weighting approach improved the classification performance.

Keywords: *Weighting Permission-based Analysis, Ensemble Extra Tree, Machine Learning, Malware Android Classification*

1. INTRODUCTION

The increasing usage of Android devices and their applications has led to increase spread of malicious applications that steal sensitive private information such as contact lists, text messages, photos, geolocations, and users' accounts through various means, including accessing information without user consent through networks. Therefore, improving the ability to detect malware on mobile devices is of paramount importance. In many applications of using machine learning, the size of a dataset is important. For example, the dataset with big size and many features will not perform well without eliminating redundant and irrelevant attributes. There are many permissions requested by applications during installation and run time; and some could be invoked by malware and non-malware applications [1]. When the number of permissions features is large, feature selection methods are used to reduce the quantity of features and improve the performance and efficiency of learning model. However, representing all features with (Boolean values) presume that all features are equally important. This is not always the case since some permissions have stronger influence in malware classification while others have less impact [2] [3]. For instance, the SMS permissions are often used in malicious applications but are used less often in benign ones. Therefore, SMS-related information has a strong impact on Android apps classification. Here, computing feature weight will help in determining the features with high influence in distinguishing android apps to malware or benign. In this work, we propose a framework of ranking features using information gain and weight-based selected ranked features using ensemble extra tree algorithm [4].

The paper is organized as follows: Section II, gives some reviews about previous works, Section III introduces the proposed research methodology, Section IV presents Results and discussions, and finally Section V provides the conclusion.

2. RELATED WORK

Generally, feature selection methods assign a binary weight to features (0 and 1) in which 1 means the feature is selected and 0 otherwise. However, feature weighting assigns a value, usually in the interval [0, 1] or [-1, 1], to each feature [5]. And representation of feature vector plays an important role in classifier performance [6]. Thus, many studies have been conducted in assigning appropriate feature weights for android malware classification to get better insight about data and obtain better classification performance. For example, the study done by [7] used Term Frequency-Inverse Document Frequency (TF.IDF) weighting method to assign weight to permission features. They tested their method on dataset of 199 malware and 200 benign applications with 330 permissions features, and evaluated their approach using K-nearest Neighbor (KNN) and Naïve Bayes (NB) algorithms. Using weighting method enhanced the performance of KNN Algorithm with 2% and 7% improvement in the NB algorithm. The authors in [8] proposed Permission Grader System (PGS) to rank the permissions based on their risk. Then they used Term Frequency-Inverse Document Frequency (TF-IDF) to score permissions and eliminate the features that are less important. Deep Neural Networks, SVM and DT were used to evaluate their proposed method. The work of [9] used Score-Based Features Selection based on static permissions as a light-weight approach to detect malware. They categorized the permissions asked by different types of malware. The study done by [2] [3] [5] [6] indicated that feature weights enhance the performance of Android malware detection and achieve better results than the same classifiers without weights of features. They used static permissions as features for the used dataset. Moreover, the study done in [10] used ensemble extra tree to assign weight to permissions in ordered to classify malware apps to their family type while in this study the android apps are classified to malware or non-malware based on weighting features using ensemble extra tree. They found that using weighing approach enhanced the classification performance. However, using (TF-IDF) methods for features weighting is computationally expensive because there are many features that occur many times. For instance, SMS-related permissions, may appear several times in one application's source code but may have a low frequency overall in the source code file [2]. Most of the studies used static permissions, while in this study we used static and hybrid permissions (static & dynamic) permissions to differentiate between malware and non-malware android applications based on feature weighting approach with ensemble extra tree algorithm. Using hybrid features make features more robust and improve the classification accuracy [11] [12]. The features are assigned weights based on the work of [10] as illustrated in the following section.

3. METHODOLOGY

In this section, we present the detailed methodology and results of the experiment. A 10-fold cross-validation is applied for the evaluations, which means that the classifiers were each executed 10 times to ensure that every portion of a split dataset was used. The average values of the 10-fold cross-validation experiments were calculated as the final results for the accuracy result. The dataset used and the proposed method are explained in details in this section.

3. 1 Dataset

The new version of dataset used in the study of [13] was used in our experiment. The dataset has 25458 samples (8643 malware apps & 16815 benign apps) with 173 permissions features (99 static permissions and 74 dynamic permissions), where each feature represents the permission. The occurrence of permission is represented by one (1) while the absence of permission is denoted by zero (0). The static permissions collected at installation time are denoted by (S) while the dynamic permissions collected at runtime are indicated by (D). Those permissions were distributed among 30 categories of the apps. The dataset is a publicly available from the website that is described in their paper [14]. From this point onwards, we refer the dataset as Hybrid dataset since it includes both static and dynamic permissions as hybrid features.

The second dataset is used in the study of [15].This dataset consists of 398 applications (199 malware and 199 normal apps) with 329 permissions features. The data set is publicly available at Kaggle website [16]. The permissions are extracted at installation time which means that the features are indicated as static permissions. The dataset will be referred as Kaggle dataset.

3.2 The proposed Model

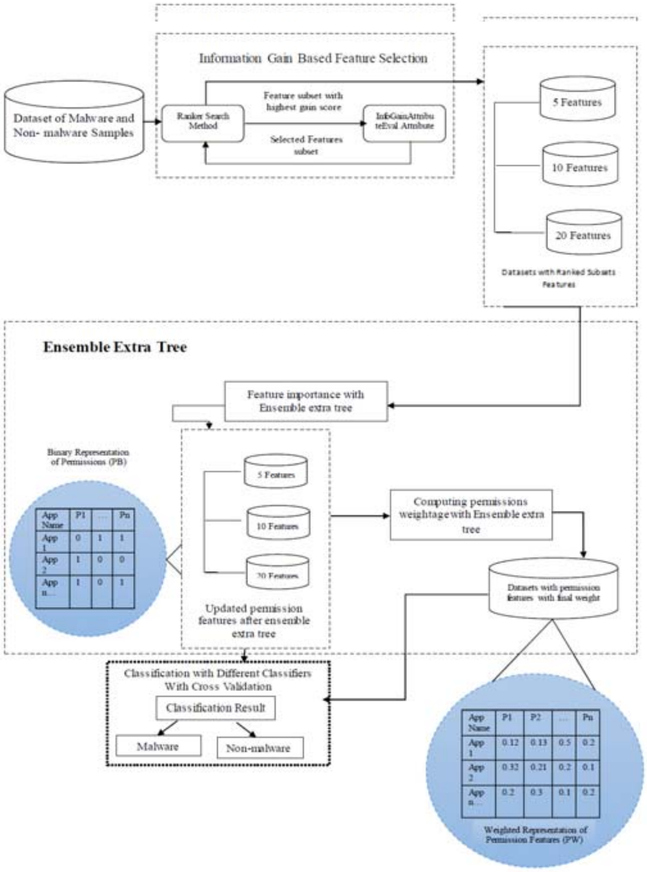
The structure and process of the proposed method with feature weights are depicted in Figure 1. First, permissions attributes are ranked using Information Gain (IG) as a feature selection method. Second, the ranked features obtained using (IG) are assigned weightage by implementing ensemble extra tree on the ranked features as shown in Figure 1 to produce feature subsets that best represented the attribute properties. The final features subsets were used to update the sets of observations with 5, 10 and 20 top ranked features respectively. Then, weightage on permission features was computed using ensemble extra trees on the updated datasets to produce a permission feature model. Here, the model is constructed by weighted representation of permission features with three different representation variations which are permission features made up from top ranked 5, 10 and 20 features respectively. The permission models were assessed by a number of machine learning classifiers to evaluate the model performance at permission representation classification.

3.2.1 Information gain

Information gain (IG) is a feature selection method that measures the entropy or (IG value) for every feature in the dataset. The larger the IG value is, the more information the feature contains. In our proposed method, the features with a higher information gain value are selected, and ranked with descending order using Ranker Search Method as shown in Figure 1. The top 5, 10 and 20 features with the highest information gain value are selected whereas the features with a lower score are removed. The final features subset is used to update datasets of best 5, 10 and 20 ranked features respectively for both dataset used (Kaggle and Hybrid dataset). Then, ensemble extra tree method is applied on those datasets with the best-ranked features 5, 10, and 20.

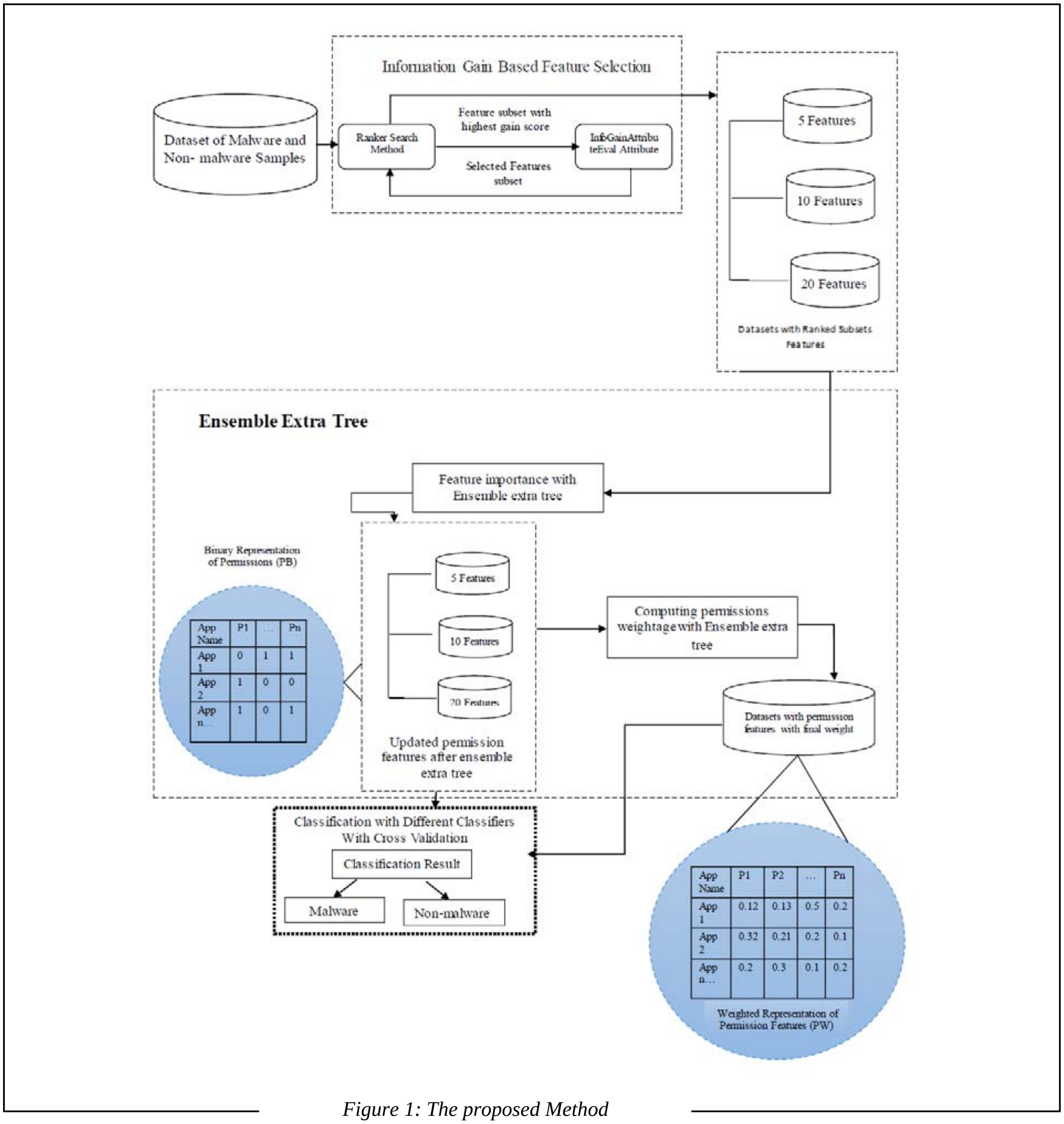
3.2.2 Feature importance with ensemble extra tree

The feature importance property that comes with Tree Based is used to choose the features with the higher score as done in the previous study [17] and in this work this is carried out by using ensemble extra tree to help find the intrinsic information of the permission features from here, new updated datasets are created from the permissions with feature subsets of 5, 10 and 20; and the features are represented with binary value means (1, 0) values as described in Figure 1. And the permissions using this approach are called binary permissions where permissions are


Information Gain Based Feature Selection
Dataset of Malware and Non- malware Samples
Ranker Search Method
Feature subset with highest gain score
InfoGainAttributeEval Attribute
Selected Features subset
5 Features
10 Features
20 Features
Datasets with Ranked Subsets Features
Ensemble Extra Tree
Feature importance with Ensemble extra tree
Binary Representation of Permissions (PB)
App Name
P1
...
Pn
App 1
0
1
1
App 2
1
0
0
App n...
1
0
1
5 Features
10 Features
20 Features
Updated permission features after ensemble extra tree
Computing permissions weightage with Ensemble extra tree
Datasets with permission features with final weight
Classification with Different Classifiers With Cross Validation
Classification Result
Malware
Non-malware
App Name
P1
P2
...
Pn
App 1
0.12
0.13
0.5
0.2
App 2
0.32
0.21
0.2
0.1
App n...
0.2
0.3
0.1
0.2
Weighted Representation of Permission Features (PW)

3.2.4 Classifier evaluations

To evaluate the performance of the proposed Model, a 10-fold cross-validation was applied for the evaluation, which means that the classifiers were each executed 10 times to ensure that every portion of a split dataset was seen. The machine learning classifiers used are (Support Vector Machine (SVM), k- Neighbors Classifier (KNN), Decision Tree (DT), Random Forest (RF), Extra Tree (EX), Naïve Bayes (NB, Linear Discriminant Analysis(LDA), Multilayer Perceptron (MLP) and Logistic Regression (LR). All the classifiers are applied on the datasets of features subsets of 5, 10 and 20 best-ranked features permissions where permissions are represented (in binary and weighted structure) as explained in Figure 1. The accuracy result of the two forms of datasets (binary and weighted representation of


Information Gain Based Feature Selection
Dataset of Malware and Non- malware Samples
Ranker Search Method
Feature subset with highest gain score
InfoGainAttributeEval Attribute
Selected Features subset
5 Features
10 Features
20 Features
Datasets with Ranked Subsets Features
Ensemble Extra Tree
Feature importance with Ensemble extra tree
Binary Representation of Permissions (PB)
5 Features
10 Features
20 Features
Updated permission features after ensemble extra tree
Computing permissions weightage with Ensemble extra tree
Datasets with permission features with final weight
Classification with Different Classifiers With Cross Validation
Classification Result
Malware
Non-malware
Weighted Representation of Permission Features (PW)

App Name	P1	…	Pn
App 1	0	1	1
App 2	1	0	0
App n…	1	0	1

App Name	P1	P2	…	Pn
App 1	0.12	0.13	0.5	0.2
App 2	0.32	0.21	0.2	0.1
App n…	0.2	0.3	0.1	0.2

Figure 1: The proposed Method

Permissions features) is illustrated in Table 1 & Table 2. The results are explained in the following section.

The comparison between two approaches of permissions representation (binary permissions and weighted permissions) have been done to evaluate the weighted approach.

4. RESULTS AND DISCUSSION

We conducted two experiments with two variance of datasets as mentioned in the previous sections. The first experiment computes the accuracy of the three groups of the datasets of 5, 10 and 20 features with binary values of permissions, whereas the second experiment calculates the accuracy rate with same group of features but with weighted representation of permissions instead of binary values .The average values of the 10-fold cross-validation was used as the final accuracy of the used classifiers.

4.1 Features Subsets of kaggle Dataset

The top 5 features with their weight scores of this dataset are descried in Figure 2. As observed from Figure 2, (android.permission.READ_PHONE_STATE) permission is the most important feature with score of (0.6145), that same permission is also assigned the highest risk value in the study of [18] based on information gain. The usage of that permission in malware apps is 0.931 while the usage in benign apps is 0.222 which indicated that permission is more critical permission to identify malware apps. Meanwhile, all the other permissions in Figure 2 are categorized as dangerous permissions as reported in the work of [13] [19] [20], these permissions were requested by most malignant apps. The top 10 permissions are shown in Figure 3. As previously described from Figure 2 below, the top 5 features occur again with slightly different order except for the first two permissions('android.permission.READ_PHONE_STATE' and android.permission.READ_SMS') in which they still dominate the top rank as in top 5 with weight of (0.4791, 0.1407) respectively.

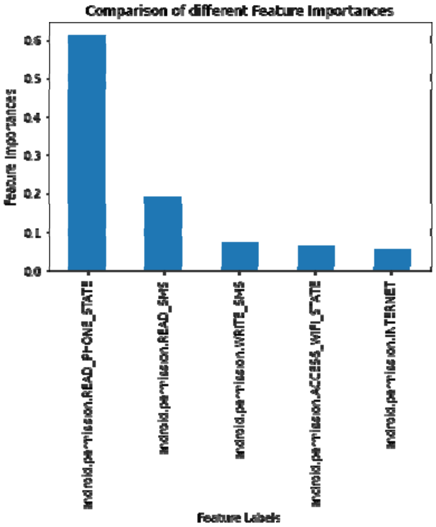


Figure 2:The top 5 Important Features with their Weights for Kaggle Dataset

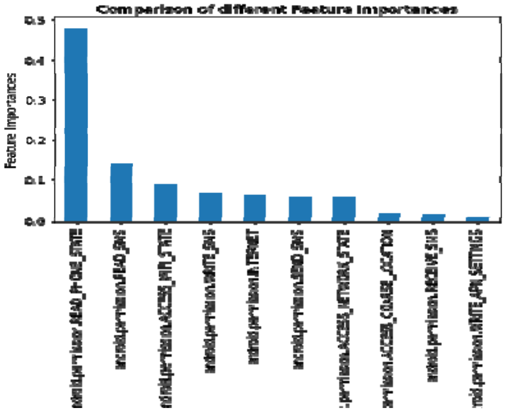


Figure 3: The top 10 Important Features with their Weights for Kaggle Dataset

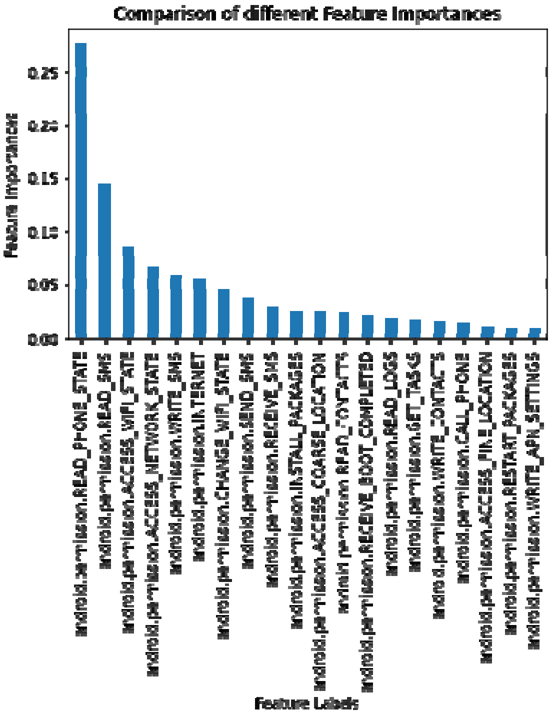


Figure 4: The top 20 Important Features with their Weights for Kaggle Datase

As observed from Figure 4 above, The result of extracting the 20 significant permissions for Kaggle Dataset using ensemble extra tree declared that the three permissions:

(android.permission.READ_PHONE_STATE, android.permission.READ_SMS, android.permission.ACCESS_WIFI_STATE) dominate the top critical features, however, the weights of 20 permissions decreased since the distribution of data has changed. For instance the weight of (android.permission.READ_PHONE_STATE) permission in the top 5 is (0.6145) while in the top 20 is (0.2769), however, that permission still ranks as the first important feature in the top 20 as shown in Figure 4. As indicated from Figures above, (READ_PHONE_STATE and READ-SMS) permissions ranked in the top of 5, 10 and 20 permissions features for Kaggle Dataset. That permissions features are more critical features in classifying malignant samples from benign samples. And they occur mostly in malicious applications and caused financial cost as reported by [19] [21].

4.2 Features Subsets of Hybrid Dataset

The most five important permissions extracted using the proposed method for Hybrid dataset is described in Figure 5. As noticed from Figure 5, the most features present are static permissions. The (Default: write contact data (S)) permission ranks the first permission with weight of (0.3895). (Default: read phone state and identity (S)) permission comes in the second rank with (0.3132) score.

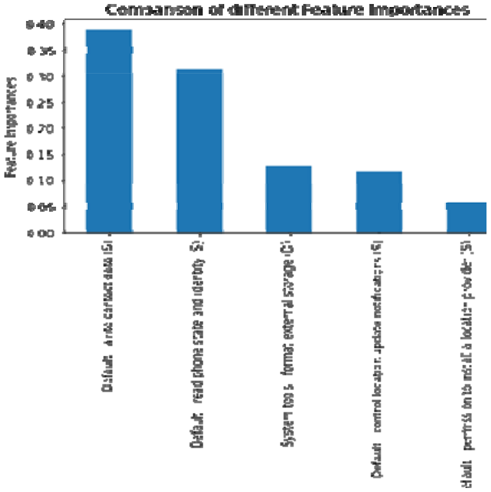


Figure 5: The top 5 Important Features with their Weights for Hybrid Dataset

The top 10 features are shown in Figure 6 where (Default: write contact data (S)) permission again dominates the top of permissions features with weight of (0.2432). As observed from the Figure 6, the most important features are static with average of (0.6).

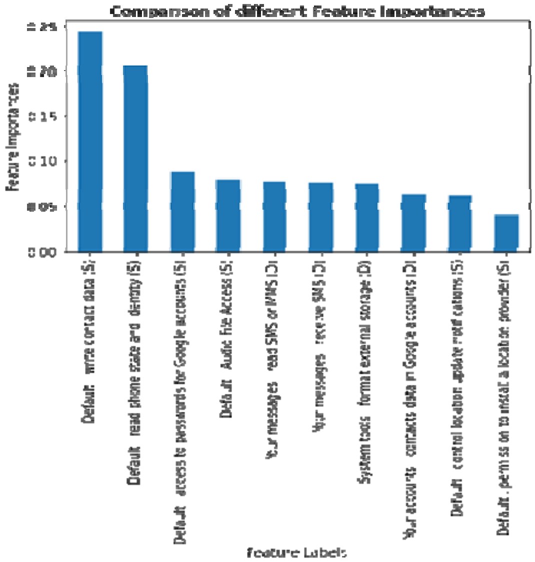


Figure 6: The highest 10 important features with their Weights for Hybrid Dataset

While the highest 20 features are depicted in Figure 7. It is displayed also that the most featured extracted are static permission which also were mostly requested by most malware as reported by the study in [13].

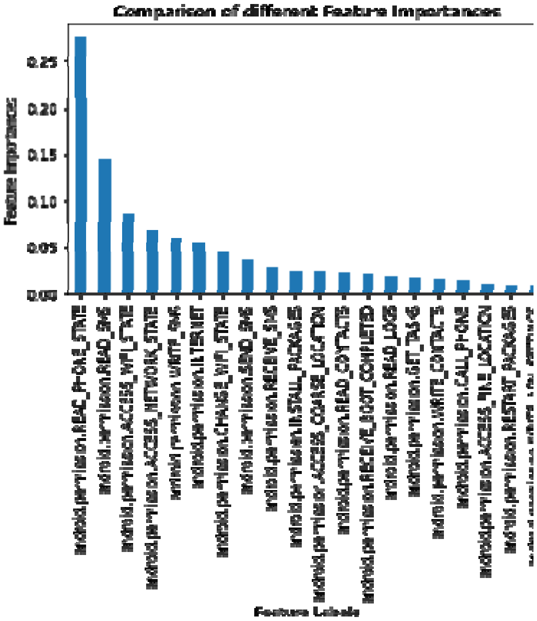


Figure 7: The highest 20 important features with their Weights for Hybrid Dataset

As noticed from Figures above for Hybrid dataset, the permissions (write contact data (S) and Default: read phone state and identity (S)) come in the first rank of 5, 10 and 20 list permissions.

That permissions are considered as dangerous permissions since they are requested by many malicious apps [19] [21].

For example, (write contact data (S)) permission under Contact pattern permission group represents the most significant feature with a score of (0.3895) comes in the first rank of the top 5 ranked features for Hybrid dataset. Meanwhile, that permission allows an application to write the user's contacts data [22]. This permission is dangerous because it involves the user's private information, as declared in [18]. The static permission (Default: read phone state and identity (S)) ranked the second in the top of 5 list with a score of (0.3132). That permission allows only access to phone state, including the phone number of the device, current cellular network information, the status of any ongoing calls, and a list of any Phone Accounts registered on the device. The study done by [18] categorized that permission as the top risky permissions with the third top risk score that identify malwares.

4.3 Classification Result

The result of the proposed method for Kaggle dataset is displayed in Table 1 below. Nine machine learning classifiers were used to assess the proposed permission models as described in section 3.2.4.

Based on Table 1 results, the performance of classification accuracy of the dataset with 20 subset features is enhanced with our proposed weighted representation of permission features approach with DT, RF, and MLP algorithms classifiers. However, the accuracy results of 5 and 10 features are similar whether features are represented in binary or weighted vector.

The results of the ranked features using integrated Information Gain (IG) and the ensemble extra tree of Hybrid dataset (static and dynamic) are explained in Table 2. As observed from Table 2, the accuracy rate increased slightly with 20 features represented in weight approach with NB and EX while the accuracy results of 5 and 10 features are similar in binary and weight representation of permissions features.

From our result obtained, we concluded that the accuracy rate is improved with weighted permissions representation approach compared to binary permissions representation approach. The highest accuracy rate achieved with DT, RF and MLP classifier algorithms for Kaggle dataset with 92%, 92 % and 93% rates respectively. For Hybrid dataset, the highest accuracy attained is with NB and EX classifier algorithms with rate of 89.68 % and 91.05 % respectively.

As observed from our findings that integrated information gain (IG) with extra tree enhanced the performance accuracy when the features permissions are represented in weightage format. In addition, the top 20 features for both datasets used (kaggle & Hybrid) achieved better results comparing to other features subsets, that results indicated that adding features (20 features) in the feature space captures the salient information variability in the feature vectors of instances belonging to the class and thus improving the classification performance. Moreover, when we comparing the results of kaggle dataset and Hybrid dataset, we found out that kaggle dataset achieved better result than Hybrid dataset that because kaggle dataset has balance class means (199 malware and 199 non-malware). So we can realized that balance dataset affect the classification results as well.

Form overall results, we can conclude that assigning weights to features provides insight about the influence of important of features in classifying apps. Some features has relative influence in identifying malware from non-malware apps, for example, (android.permission.READ_PHONE_STATE) permission in Kaggle dataset has the most influence in classifying android apps while (android.permission .INTERNET) permission has the lowest influence. And representing the features with Boolean values will treat all features similarly [2] [3]. Therefore, it is important to assign weight to features that are more related to class object. Moreover, representing features with weight values improved the classification performance as proved in our results [2] [3] [6].

Table 1: Classification Accuracies of the Tested Machine Learning Algorithms for Different Representation of Features (binary & weight) for Kaggle Dataset

Classifier Algorithm Used	**Accuracy for Datasets with binary representation of permissions features approach**			**Accuracy for Datasets with weight representation of permissions features approach**		
	5 features	**10 features**	**20 features**	**5 features**	**10 features**	**20 features**
Svm	0.92	0.92	0.92	0.92	0.92	0.92
knn	0.80	0.92	0.92	0.80	0.92	0.91
NB	0.89	0.84	0.86	0.89	0.84	0.86
DT	0.93	0.92	0.91	0.93	0.92	**0.92**
RF	0.93	0.92	0.91	0.93	0.92	**0.92**
EX	0.93	0.92	0.92	0.93	0.92	0.92
(LDA)	0.92	0.92	0.92	0.92	0.92	0.92
MLP	0.93	0.93	0.92	0.92	0.92	**0.93**
(LR)	0.93	0.92	0.92	0.92	0.92	0.92

Table 2: Classification Accuracies of the Tested Machine Learning Algorithms for Different Representation of Features (binary & weight) for Hybrid dataset

Classifier Algorithm Used	**Accuracy for Datasets with binary representation of permissions features approach**			**Accuracy for Datasets with weight representation of permissions features approach**		
	5 features	**10 features**	**20 features**	**5 features**	**10 features**	**20 features**
Svm	0.84	0.88	0.9014	0.84	0.87	0.8721
knn	0.34	0.34	0.6584	0.34	0.34	0.6571
NB	0.84	0.88	0.8962	0.84	0.88	**0.8968**
DT	0.84	0.90	0.9102	0.84	0.90	0.9101
RF	0.84	0.90	0.9106	0.84	0.90	0.9107
EX	0.84	0.90	0.9104	0.84	0.90	**0.9105**
LDA	0.83	0.85	0.8723	0.83	0.85	0.8723
MLP	0.84	0.90	0.9110	0.84	0.88	0.9029
LR	0.84	0.88	0.9017	0.84	0.85	0.8517

5. CONCLUSION AND FUTURE WORK

In this paper, we propose a method to classify Android samples to malware or non-malware based on feature weighting approach. The proposed model assigns weight of the ranked features obtained after using Information Gain (IG) selection method with Extra tree algorithm. Extra tree is applied on the datasets of top 5, 10 and 20 ranked features obtained using Information Gain (IG). Then that subsets of that features are assigned weight based on importance score obtained using extra tree classifier algorithm.

Nine machine learning classifiers were used to assess the proposed permission models (binary and weighted structure). The results show that datasets with 20 weighed features subset achieved the highest accuracy with DT, RF and EX classifiers for kaggle dataset comparing with the results of features represented with binary structure and achieved good results with EX and NB classifiers for Hybrid dataset compared to the results of features represented with binary structure . Our experiments verify that weight representation of the permissions feature approach contributes in improving classification result by selecting the most important discriminative features that have the most impact on classifying android apps to malware or non-malware.

REFERENCES:

[1] Kumar S, Viinikainen A, Hamalainen T. A network-based framework for mobile threat detection. Proc - 2018 1st Int Conf Data Intell Secur ICDIS 2018. 2018; 227–33.

[2] Xu Y, Wu C, Zheng K, Wang X, Niu X, Lu T. Computing Adaptive Feature Weights with PSO to Improve Android Malware Detection. Secur Commun Networks. 2017; 2017.

[3] Cai L, Li Y, Xiong Z. JOWMDroid: Android malware detection based on feature weighting with joint optimization of weight-mapping and classifier parameters. Comput Secur [Internet]. 2021; 100:102086. Available from: https://doi.org/10.1016/j.cose.2020.102086.

[4] Geurts P, Ernst D, Wehenkel L. Extremely randomized trees. Mach Learn. 2006 Apr; 63(1):3–42.

[5] Aggarwal CC. An introduction to data classification. Data Classification: Algorithms and Applications. 2014. 1-36 p.

[6] Vinod P, Zemmari A, Conti M. A machine learning based approach to detect malicious android apps using discriminant system calls. Futur Gener Comput Syst [Internet]. 2019; 94:333–50. Available from: https://doi.org/10.1016/j.future.2018.11.021

[7] Sahin DO, Kural OE, Akleylek S, Kilic E. New results on permission based static analysis for Android malware. 2018 6th Int Symp Digit Forensic Secur [Internet]. 2018; 1–4. Available from: https://ieeexplore.ieee.org/document/8355377/.

[8] Dharmalingam VP, Palanisamy V. A novel permission ranking system for android malware detection—the permission grader. J Ambient Intell Humaniz Comput. 2020 Apr 20.

[9] La C, Mar K. Permission-based Feature Selection for Android Malware Detection and Analysis. Int J Comput Appl. 2018; 181(19):29–39.

[10] Alswaina F, Elleithy K. Android Malware Permission-Based Multi-Class Classification Using Extremely Randomized Trees. IEEE Access. 2018; 6:76217–27.

[11]] Nisa M, Shah JH, Kanwal S, Raza M, Khan MA, Damaševičius R, et al. Hybrid malware classification method using segmentation-based fractal texture analysis and deep convolution neural network features. Appl Sci. 2020; 10(14).

[12] Martín A, Lara-Cabrera R, Camacho D. Android malware detection through hybrid features fusion and ensemble classifiers: The AndroPyTool framework and the OmniDroid dataset. Inf Fusion. 2019 Dec 1; 52:128–42.

[13] Mahindru A, Singh P. Dynamic Permissions based Android Malware Detection using Machine Learning Techniques. Proc 10th Innov Softw Eng Conf - ISEC '17 [Internet]. 2017 ;(March 2018):202–10. Available from: http://dl.acm.org/citation.cfm?doid=3021460.3021485

[14] A. Mahindru, "Android Malware and Normal permissions dataset," 2018. [Online]. Available: https://data.mendeley.com/datasets/958wvr38gy/5.

[15] Lopez CCU, Cadavid AN. Machine learning classifiers for android malware analysis. 2016;

[16] https://www.kaggle.com/xwolf12/datasetandroidpermissions

[17] Alkaaf HA, Ali A, Shamsuddin SM, Hassan S. Exploring permissions in android applications using ensemble-based extra tree feature selection. Indones J Electr Eng Comput Sci. 2020; 19(1):543.

[18] Deypir M. Entropy-based security risk measurement for Android mobile applications. Soft Comput. 2019 Aug 1; 23(16):7303–19.

[19] Dini G, Martinelli F, Matteucci I, Petrocchi M, Saracino A, Sgandurra D. Risk analysis of Android applications: A user-centric solution. Futur Gener Comput Syst. 2018 Mar 1; 80:505–18.

[20] Alzaylaee MK, Yerima SY, Sezer S. DL-Droid: Deep learning based android malware detection using real devices. Comput Secur. 2020; 89.

[21] Abdulla S, Altaher A. Intelligent approach for android malware detection. KSII Trans Internet Inf Syst. 2015; 9(8):2964–83.

[22] Kumar R, Zhang X, Wang W, Kumar JAY, Sharif A. A Multimodal Malware Detection Technique for Android IoT Devices Using Various Features. 2020