Düzce University Journal of Science & Technology

Research Article

Performance Analysis of Machine Learning Algorithms for Malware Detection by Using CICMalDroid2020 Dataset[1]

Yusuf SÖNMEZ[a], Meltem SALMAN[b,*], Murat DENER[b]

[a] *Department of Computer Technologies and Cyber Security, Faculty of Information and Telecommunication Technologies, Azerbaijan Technical University, Baku, AZERBAIJAN*
[b] *Department of Information Security Engineering, Graduate School of Natural and Applied Sciences, Gazi University, Ankara, TURKEY*
** Corresponding author's e-mail address: meltem.salman@tai.com.tr*
DOI: 10.29130/dubited.1018223

ABSTRACT

In parallel with the developments in technology, access to information has become easier. Although this situation has a positive effect on our lives, it is an inevitable fact that information has become a target by malicious people. Theft of information and its use as a threat by these people have caused concerns about information security. Malware developed for these purposes poses a great danger to the security of information. In the face of this situation, which increases as access to information becomes easier, researchers have accelerated their work on detecting and preventing malware and ensuring information security. In the literature, it is seen that the detection of malicious software has been carried out with different studies. In this study, malware detection was carried out using the WEKA program. The effects of different machine learning classifiers, feature extraction and the parameters that affect the performance of the classification that gives the best result were examined in the analyzes made with the CICMalDroid2020 dataset. The results are presented in detail.

Keywords: *Malware detection, CICMalDroid2020, WEKA, Machine learning*

CICMalDroid2020 Veri Kümesi Kullanılarak Kötü Amaçlı Yazılım Tespiti için Makine Öğrenimi Algoritmalarının Performans Analizi

ÖZ

Teknolojideki gelişmelere paralel olarak bilgiye erişim kolaylaşmıştır. Bu durumun hayatımıza pozitif etkisi olsa da bilginin hedef haline geldiği kaçınılmaz bir gerçektir. Kötü amaçlı kişiler tarafından bilgilerin çalınması, tehdit unsuru olarak kullanılması bilgi güvenliği konusunda endişelere sebep olmuştur. Bu amaçlarla geliştirilen kötücül yazılımlar, bilginin güvenliği açısından büyük bir tehlike oluşturmaktadır. Bilgiye erişim kolaylaştıkça artan bu durum karşısında araştırmacılar, kötücül yazılımların tespiti, engellenmesi ve bilgi güvenliğinin sağlanması konusunda çalışmalarına hız kazandırmışlardır. Literatürde, farklı çalışmalar ile kötücül yazılımların tespiti gerçekleştirildiği görülmektedir. Bu çalışmada ise, kötücül yazılım tespiti WEKA programı kullanarak gerçekleştirilmiştir. CICMalDroid2020 veri seti ile yapılan analizlerde, farklı makine öğrenmesi sınıflandırıcılarının, özellik çıkarımının ve en iyi sonucu veren sınıflandırmanın performansını etkileyen parametrelerin etkisi incelenmiştir. Sonuçlar, detaylı bir şekilde aktarılmıştır.

Anahtar Kelimeler: *Kötücül yazılım tespiti, CICMaldroid2020, WEKA, Makine öğrenimi*

[1] The part of this study was presented as an oral presentation in ICAIAME 2021.
Received: 03/11/2021, Revised: 12/12/2021, Accepted: 17/12/2021

I. INTRODUCTION

Information is a concept that is easier to access today and becomes a target, as it gets easier. This situation causes the threat of information and once again brought the importance of information security to the agenda. One of the remarkable issues of in the last decades is the malware detection for Android systems. It is seen that there are many studies in the literature to ensure the security of information and to detect malicious approaches. Some of these are studies using machine learning. Some studies in the literature are mentioned below.

Martin et al. [1] examined upwards of 80 000 applications for Android system that were tagged as malware with at least one antivirus (AV) engine, and about 260 000 malware signatures. In their study, 41 malware families were identified and relationships between these were examined. Based on these relationships, they have seen that some are marked as 'Unknown' while many of them are labeled as Adware truly hazardous malicious applications. It is stated that thanks to Graphics Community Algorithms and Machine Learning, such Unknown applications can further unify dissimilart AV detections to classify as Adware or Malicious risks with the use of Machine Learning and Graphics Community Algorithms.

Wu et al. [2] have proposed a system for Android to detect malwares that offers correct classification and sensitive data transfer analysis by using machine learning approach. A detailed analysis was carried out to extract API-level features related to the data flow and to improve the nearest neighbor classification model. Using a total of 2210 samples, 1160 of which were benevolent and 1050 were malicious, they concluded that their system had an accuracy of 97.66%.

Martinelli et al. [3], based on the malware HummingBad, performed malware detection analysis for Android with two different methods. The first method is machine learning. The other method is model control based approach. These two methods evaluated the results.

Surendran et al. [4] proposed a hybrid malware detection system which bases Tree Augmented Naive Bayes (TAN). They used conditional dependencies between related dynamic and static features (system and API calls, permissions) needed for an application's the functionality. Three ordered logistic regression classifiers are trained, corresponding to system and API calls, and permissions. Their output were modeled as TAN for determination if the applcation is malicious. They concluded that the suggested system could detect malwares with 0.97 accuracy.

Razgallah et al. [5] investigated malware detection for Android applications by using fundamental mechanisms and approaches. They identified the advantages and disadvantages of each approach and proposed research paths to advance knowledge on this topic.

Wang and Li [6] used weight-based detection (WBD) to categorize and understand the characteristics of Android malware and harmless applications. They examined 112 core attributes of executing task data structure in Android system and evaluated detection accuracy with a set of datasets of various sizes.

Milosevic et al. [7] presented two machine learning supported approaches to perform static analysis of Android malware. They based on permissions and source code analysis (by utilizing the word bag representation model). For both approaches, they reached 95.1% F-score and 89% F-measurement for two classification models, source code-based and permission-based, respectively.

Bai et al. [8] studied about which approach is better to classify for malware family. For three common Android malware datasets, five multiclassification methods were designed for prediction of the Android malware family. They have created 250 common features which shared by Android malware. Also, they investigated transfer learning effect for adapting the model to three malware datasets.

Rehman et al. [9] offered a hybrid method to detect malware in Android applications. It is stated that for Android Applications, the suggested method considers both heuristic and signature based analysis. Reverse engineering were used for extraction of binaries and manifest files. Machine learning algorithms such as classifiers such as as Decision Tree, SVM, KNN, and W-J48 were preferred for malware detection. SVM for binaries and KNN for manifest.xml files have been found to be the most suitable options. It was concluded that the proposed hybrid model reached improved accuracy to detect malware.

Chen et al. [10] machine learning methods and combined network traffic analysis to detect malware. They said that a small portion of the network traffic created by malicious applications is dangerous and most of it harmless. For this reason, they found that when the traffic model leans towards modeling innocuous traffic, it causes an unstable data problem and used unbalanced classification methods including SVM (support vector machine) and SMOTE (synthetic minority oversampling technique), SVMCS (SVM cost sensitive) and C4.5CS (C4.5 cost sensitive)to solve the problem. At the same time, they proposed to utilize the imbalanced data gravity-based classification (IDGC) algorithm for classification unbalanced data in order to avoid performance degradation.

In this paper, analyzes were carried out with machine learning using WEKA software for the detection of malware. In the second part, there is a literature review. The next one includes dataset, the mathematical expression of the evaluation criteria and the WEKA program are given. In the fourth part, the studies carried out and the findings obtained are discussed in detail. The comparison of machine learning classifiers, the effect of feature extraction, the effects of distance criterion and number of neighborhoods for KNN classifier are examined. In the last section, the studies are briefly evaluated.

II. MATERIAL AND METHOD

In this paper, the data set CICMalDroid2020 [11] (CIC: Canadian Institute for Cybersecurity) was used. Mahdavifar et al. [11] stated that they collected more than 17341 samples from different sources including Contagio security block, MalDozer, VirusTotal, AMD datasets during the dataset collection phase. Because of their analysis, they determined the number of properly working samples as 13077 out of 17341 samples. After the data collection process, which lasted from December 2017 to December 2018, they categorized the data according to whether it is malware or not, and if it is malicious, what type it is. As a result, they obtained a set containing 11598 data with a total of 5 categories. The dataset has 9803 malware, including Adware, Banking Malware, SMS Malware, and Mobile Riskware. Its benign number is 1795 and it is in the benign software category. In addition, they presented two different sets of 470 and 139 extracted features. In this study, the authors' datasets with 470 extracted features were used. Detailed information about the data set is given in Table 1.

Table 1. *CICMalDroid2020 dataset categories.*

Adware	Banking Malware	SMS Malware	Mobile Riskware	Benign	Total
1253	2100	3904	2546	1795	11598

The analysis of the dataset using machine-learning classifiers was carried out with the WEKA program which was developed at the University of Waikato. It is abbreviation for Waikato Environment for Knowledge Analysis. This code, which is a JAVA open source library, contains an algorithm that can be applied to devices with Android operating system [12]. In the classification results made with WEKA, False Positive Ratio (FPR), True Positive Ratio (TPR), Precision, Recall, F-Measure etc. values are given. These values are an important criterion in interpreting the results. TPR, correctly defined data; FPR, misidentified data; Precision is expressed as the ratio of the correct data of a category to the incorrect data of that category and is formulated as follows [11],[12].

$$TPR = \frac{TP}{FN+TP} \quad (1)$$

$$FPR = \frac{FP}{TN+FP} \quad (2)$$

$$Precision = \frac{TP}{TP+FP} \quad (3)$$

$$F - Measure = \frac{2*TP}{2*TP+FP+FN} \quad (4)$$

$$Recall = \frac{TP}{FN+TP} \quad (5)$$

$$Accuracy\ (\%) = \frac{TN+TP}{TP+FN+FP+TN} * 100 \quad (6)$$

The path followed in the study is given in Figure 1. Up to this point, the dataset and WEKA evaluation criteria are included. The next steps are covered in the 'Results and Discussion' section in detail.

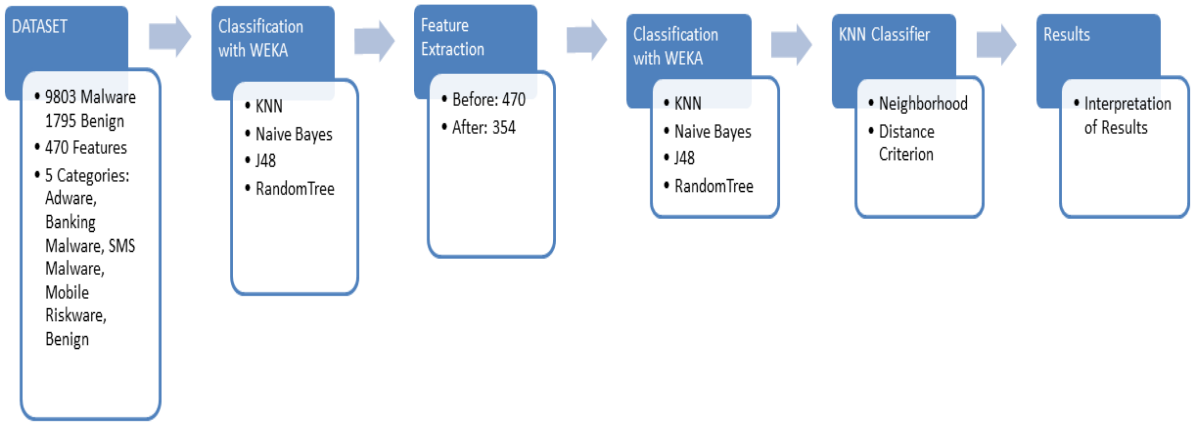

Figure 1. *Flow chart of study.*

III. RESULTS AND DISCUSSION

In this study, in which analyzes were carried out for the detection of malicious software, primarily the dataset was provided. Then, the CICMalDroid2020 dataset with 470 features was first analyzed using the machine learning (ML) classifiers as KNN, Naive Bayes (NB), J48 and RandomTree (RT) algorithms. After, feature extraction was performed and the results were compared with the same ML classifiers. The effect of different parameters was examined by using the algorithm that gave the best results.

A. EFFECTS OF ALGORITHMS

When the literature is examined, it is seen that ML algorithms are frequently used in malware detection. Within this scope, 4 different classifiers, namely KNN, NB, J48 and RT, are included in the study. Classification results using the WEKA program are given in Figure 2. In the evaluation of success made using the accuracy percentage, it is seen that the KNN classifier is the algorithm that gives the best result with a success rate of 91.5%. The lowest success is the RT algorithm with 71.1%. Results for J48 and NB are 87.5% and 80.5%, respectively. In Figure 3, TPR, FPR, etc. given in WEKA analysis outputs. The results of the criteria are given. All classifiers, Category 3, SMS Malware, appear to have high accuracy. This result is in line with the findings obtained from the study [11].

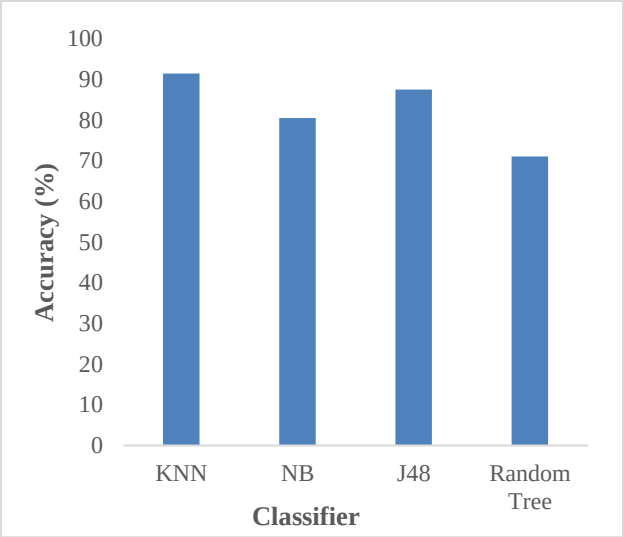

Figure 2. *Classifier accuracy.*

		TPR	FPR	PRECISION	RECALL	F-MEASURE	ACCURACY (%)
KNN	1	0.889	0.025	0.811	0.889	0.848	91.4554
	2	0.918	0.018	0.918	0.918	0.918	
	3	0.991	0.03	0.943	0.991	0.966	
	4	0.911	0.024	0.916	0.911	0.914	
	5	0.768	0.011	0.926	0.768	0.84	
NB	1	0.737	0.063	0.587	0.737	0.654	80.488
	2	0.841	0.077	0.708	0.841	0.769	
	3	0.96	0.07	0.875	0.96	0.915	
	4	0.617	0.004	0.978	0.617	0.757	
	5	0.739	0.032	0.809	0.739	0.773	
J48	1	0.735	0.036	0.71	0.735	0.722	87.4978
	2	0.852	0.019	0.91	0.852	0.88	
	3	0.983	0.044	0.919	0.983	0.95	
	4	0.841	0.027	0.898	0.841	0.869	
	5	0.813	0.033	0.82	0.813	0.816	
RANDOM TREE	1	0.429	0.046	0.533	0.429	0.476	71.0898
	2	0.747	0.062	0.727	0.747	0.737	
	3	0.941	0.135	0.78	0.941	0.853	
	4	0.688	0.076	0.719	0.688	0.703	
	5	0.397	0.058	0.555	0.397	0.463	

Figure 3. *Criteria.*

In Table 2, the numbers of data classified according to categories are given. Category 1 (Adware), Category 2 (Banking Malware), Category 3 (SMS Malware), Category 4 (Mobile Riskware) and Category 5 (Benign) are expressed as a, b, c, d and e, respectively. When the two highest results (KNN - 91.5% and J48 - 87.5%) are compared, it is seen that the number of benign software detected in J48 (1459) is higher than that detected in KNN (1379). The sum of the numbers in which each category is correctly classified is 10607 data for KNN. In J48, the correct classification result in all categories is 10181. In short, although the success of detecting benign software in J48 is 81.3% (76.8% for KNN), when the total accuracy percentage is considered, it is seen that KNN is a better classifier under these conditions.

Table 2. *Distribution of categories.*

	a	b	c	d	e	
KNN	**1114**	**23**	**33**	**47**	**36**	**a**
	48	**1927**	**56**	**40**	**29**	**b**
	6	**20**	**3867**	**10**	**1**	**c**
	89	**55**	**37**	**2320**	**45**	**d**
	117	**75**	**108**	**116**	**1379**	**e**
NB	**924**	151	77	9	92	a
	151	**1767**	90	10	82	b
	54	89	**3746**	7	8	c
	285	299	260	**1571**	131	d
	160	189	109	10	**1327**	e
J48	**921**	**47**	**39**	**100**	**146**	**a**
	99	**1790**	**107**	**48**	**56**	**b**
	11	**20**	**3836**	**16**	**21**	**c**
	149	**73**	**85**	**2142**	**97**	**d**
	117	**36**	**105**	**78**	**1459**	**e**
RANDOM TREE	**538**	140	188	195	192	a
	82	**1569**	236	115	98	b
	43	82	**3674**	56	49	c
	144	148	270	**1751**	233	d
	202	218	343	319	**713**	e

B. EFFECT OF FEATURE EXTRACTION

Feature reduction is one of the important studies in malware detection studies. In this study, analyzes were performed again by subtracting the 116 features that had the lowest effect on the ranking from the number of 470 features. The results obtained for the KNN, NB, J48 and RT classifiers were compared with the results before feature extraction (Figure 4).

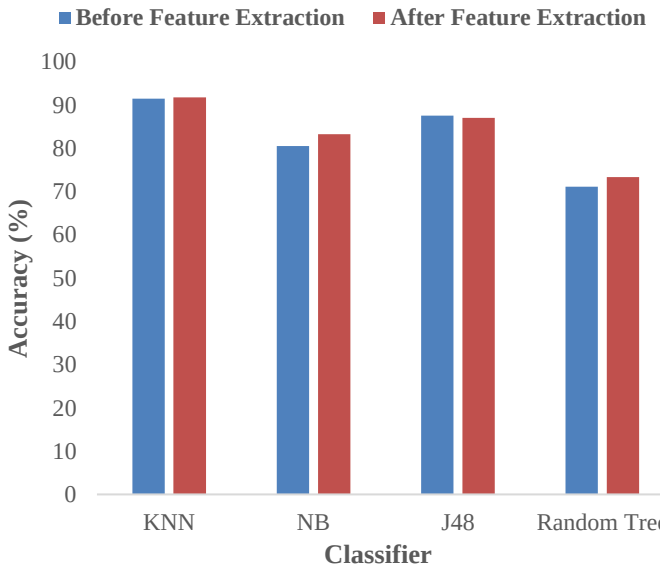

Figure 4. *Accuracy comparison before and after feature extraction.*

At Figure 4, the change in accuracy for the KNN classifier after feature extraction was minimal. The greatest increase in the accuracy of the results was observed for NB. Contrary to the others, there is a small decrease in J48.

In the analyzes made so far, the success of different classifiers and the effect of feature extraction in malware detection have been examined. In the comparison, as seen in Figure 4, the malware was tagged with the best KNN classifier. Based on this achievement, analyzes were made for the KNN classifier and 354 features in the next parts of the study.

C. EFFECT OF NEIGHBORHOOD AND DISTANCE CRITERION

According to the findings from the study, KNN is the best performing classifier in malware detection among other classifiers. Based on this information, new analyzes were made by changing the number of neighborhoods (k) and distance criteria parameters in the KNN algorithm. The aim is to observe the change of in accuracy by trying different parameters in the algorithm where the best classification is obtained.

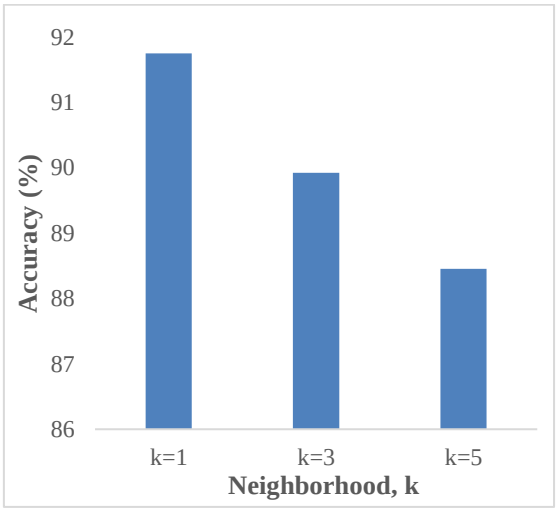

Figure 5. *Neighborhood effect.*

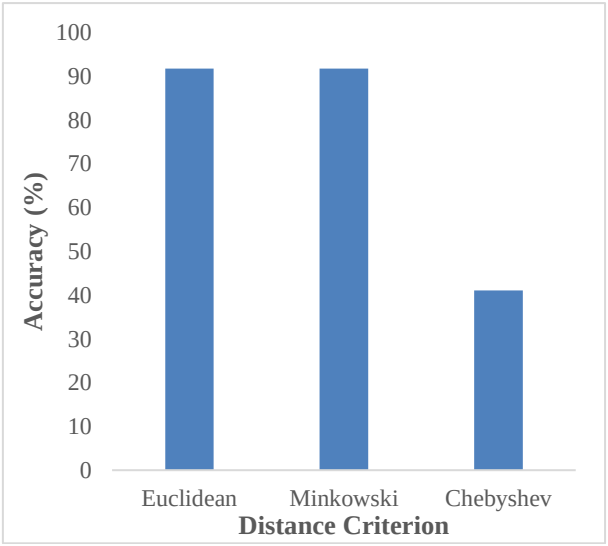

Figure 6. *Distance criterion effect.*

In Figure 5, the effect of the number of neighborhoods for the KNN classifier is examined. It is seen that the increase in k has a negative effect on performance in malware detection. Figure 6 shows the results of the distance criterion parameter in the KNN classifier for malware detection. Until this section, the Euclidean distance criterion has been used in all studies with the KNN classifier. When Minkowski and Chebyshev distance measures are used, it is observed that the results of the study with the Minkowski distance criterion and the study with the Euclidean are the same. When analyzed with Chebyshev, it was concluded that his performance was worse (about 50%) than Euclidean.

IV. CONCLUSION

The information age we live in has brought along some problems as well as providing great convenience for humanity. As the access to information, technology and internet became easier, malicious use also emerged. In parallel with the increase in these threats, which pose a great danger to information security, prevention and detection activities in these areas have also accelerated. In the field of information security, malware detection studies, which are also frequently encountered in the academic world, are conducted to identify threats developed with malicious intent. In this study, it has been tried to determine the malware, which is an important issue in terms of information security, by using the ML classifiers in the WEKA program.

In the analyzes performed using the CICMalDroid2020 dataset, the success rates of different classifiers were examined. The two best classification achievements are KNN and J48, respectively. Despite the higher detection of benign software in J48, KNN performed better overall. An increase in the percentage of success in malware detection studies is associated with accurate detection of malware. Labeling benign software as malicious software can lead to financial and time losses. However, since the labeling of malicious software as benign will cause greater damage, the most basic target is the studies for the correct labeling of malware. In this context, it is seen that malware tagging is more accurate with KNN than with J48. In addition, the effect of feature extraction in addition to classifier performance is also investigated. According to the findings obtained from these two studies, the best classification success (before and after feature extraction) belongs to the KNN algorithm. Therefore, in the next stages, the distance criterion and the change in the number of neighborhoods, which are the parameters that will affect the success performance, were examined in the KNN algorithm. It was found that the results of the Minkowski and Euclidean distance measures were the same, while the performance of the Chebyshev distance criteria was halved. Finally, it is seen that the increase in the number of neighbors does not provide the desired success in detecting malware.

Shortly, in this age where information is under threat, it is of great importance to identify and prevent these threats. The detection of software made with malicious intentions is one of the shining areas of recent years. With the study prepared within this scope, analyzes were made with the dataset containing malicious (4 different categories) and benign software, and the labeling of benign-malicious was made. There are many studies in the literature regarding the correct labeling of malware. This study, on the other hand, enriches the literature in terms of examining the effect of classifier, feature extraction in malware detection with the CICMalDroid2020 dataset and the effect of KNN parameters on performance based on the results obtained.

V. REFERENCES

[1] Martín, J. A. Hernández and S. de los Santos, “Machine-Learning based analysis and classification of Android malware signatures,” *Future Generation Computer Systems*, vol. 97, pp. 295–305, 2019.

[2] 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.

[3] F. Martinelli, F. Mercaldo, V. Nardone, A. Santone and G.Vaglini, "Model checking and machine learning techniques for HummingBad mobile malware detection and mitigation," Simulation Modelling Practice and Theory, 2020.

[4] R. Surendran, T.Thomas and S. Emmanuel, "A TAN based hybrid model for android malware detection," *Journal of Information Security and Applications*, vol. 54, 2020.

[5] A. Razgallah, R. Khoury, S. Hallé and K. Khanmohammadi, "A survey of malware detection in Android apps: Recommendations and perspectives for future research," *Computer Science Review*, vol. 39, 2021.

[6] X. Wang and C. Li, "Android malware detection through machine learning on kernel task structures," *Neurocomputing*, vol. 435, pp. 126–50, 2021.

[7] N. Milosevic and A. Dehghantanha, "Choo KR. Machine learning aided Android malware classification R," *Computers and Electrical Engineering*, vol. 61, pp. 266–74, 2017.

[8] Y. Bai, Z. Xing, D. Ma, X. Li and Z. Feng, "Comparative analysis of feature representations and machine learning methods in Android family classification," *Computer Networks*, vol. 184, 2021.

[9] Z. U. Rehman, S. N. Khan, K. Muhammad, J. W. Lee,Z. Lv, S. W. Baik, et al. "Machine learning-assisted signature and heuristic-based detection of malwares in Android devices," *Computers and Electrical Engineering*, vol. 69, pp. 828–41, 2018.

[10] Z. Chen, Q. Yan, H. Han, S.Wang, L. Peng, L. Wang, et al. "Machine learning based mobile malware detection using highly imbalanced network traffic," *Information Sciences*, 2018.

[11] S. Mahdavifar, A. F. Abdul Kadir, R. Fatemi, D. Alhadidi and A. A. Ghorbani, "Dynamic Android Malware Category Classification using Semi-Supervised Deep Learning." Proceedings - IEEE 18th International Conference on Dependable, Autonomic and Secure Computing, IEEE 18th International Conference on Pervasive Intelligence and Computing, IEEE 6th International Conference on Cloud and Big Data Computing and IEEE 5th Cyber Science and Technology Congress, DASC/PiCom/CBDCom/CyberSciTech 2020.

[12] D. Rathi and R. Jindal, "DroidMark: A Tool for Android Malware Detection using Taint Analysis and Bayesian Network," *International Journal on Recent Trends in Computing and Communication*, vol. 6, pp. 71-76, 2018.