Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 55-12 (2022) 462–467

Effect of Features Extraction and Selection on the Evaluation of Machine Learning Models

Omar HABIBI * **Mohammed CHEMMAKHA** ** **Mohamed LAZAAR** ***

ENSIAS, Mohammed V University in Rabat, Morocco

* *E-mail : omar_habibi2@um5.ac.ma*
** *E-mail : mohammed_chemmakha@um5.ac.ma*
*** *E-mail : mohamed.lazaar@ensias.um5.ac.ma*

Abstract: The exponential growth of sophisticated malware attacks against computer systems, has alerted IT security experts on the shortcomings of traditional protection tools because, they became unable to detect new families of malware that are more advanced and use advanced tools such as polymorphism, metamorphism, and obfuscation tools. Nowadays, machine learning is widely used in several IT fields; and also in cybersecurity, and can be an essential tool for malware detection, moreover, it can go beyond the limits of classic malware detection methods, such as the signature-based method, the anomaly-based method and the hybrid-based method, etc. The purpose of this study is to analyze the feature selection and extraction effects on the performance of malware classification model using machine learning. The results show that reducing dimensionality of datasets can help to improve the efficiency of security models in restricted time but with high performance, Random Forest using chi-square achieves an accuracy of 99.51%.

Keywords: Machine Learning, Malware Detection, Feature Extraction, ANN, KNN, Logistic Regression (LR), Random Forest (RF)

1. INTRODUCTION

In recent years, billions of people around the world are using internet, hence some criminals take advantage of this situation for their benefit by developing sophisticated tools to carry out cyber-attacks, which makes these attacks a serious problem for cybersecurity experts and cause serious damage to companies and to real users. Malware is a software that aims to cause damage to victims, such as causing malfunctions in a system, altering user or company data, violating access controls, etc. A malware can take the format of a benign software, of a script or an executable code. There are quite a few types of malware like Adware, Spyware, Worms, Viruses, Trojan, etc. According to Computer Economics financial [1], the loss due to malware attack has grown quadruple from $3.3 billion in 1997 to $13.3 billion in 2006 [1]. Generally, all detected malware by antivirus are recorded by their signature in special databases, new sophisticated types of malware, such as polymorphic malware and metamorphic ones and malware that rely on obfuscation tools in their operation, are not recorded in the malware databases which makes antivirus, IDS useless protection tools, and the detection of these malware by the use of classic countermeasures is almost an impossible treatment, because these new generations of malware change their signatures. There are some malware types that can also change their behavior towards protection tools and sometimes even change source code of malware to infiltrate protection systems. In view of the foregoing, we can deal with this serious situation with machine learning as a solution to such kind of malicious files.

If we are supposed to work on a large dataset with a large amount of features, machine learning models require high performance machines, and interesting learning time, hence the usefulness of the extraction and selection phase; which make it possible to eliminate redundant features, or highly correlated features, and consequently increase the performance of the model while reducing the training time, and reducing the use of computing resources. It should be mentioned that if we speak from a technical point of view, if we deploy a model that can detect malware with a minimum of features to process, this will accelerate the processing of malware detection on computer security tools, such as the famous following tools: antivirus, IDS for example, where detection time is a major factor in judging the performance of these security tools. In this work, we will treat this phase of extraction and selection of features on a dataset that contains 138048 lines and 57 features; in order to examine the effects and limits of this phase, and the modifications necessary to answer the essential problematic of our study. This paper is orginazed as follow: (1) exploring scientific research that has broached the same subject, (2) describing data preprocessing tools and the ML implemented methods, (3) analyzing the results obtained, (4) and discussing future work that can be developed for improving the results and accuracy.

10.1016/j.ifacol.2022.07.355

2. RELATED WORK

The study by Shhadat & Al-Sharif [2] is a comparative one with the study by Chumachenko, K. [3], the second study consists of a heuristic strategy through dynamic analysis and application of Random Forest on the features of the dataset that are less correlated and an implementation of machine learning models. The study of Shhadat consists of the improvement of the results using cross-validation with k=15 to balance the dataset. The researchers exploited the same dataset in both studies that consists of 1156 files, 172 clean files and 984 malware. The study by Shhadat & Al-Sharif shows higher accuracy for multiclass classification and binary classification. In the study by Chumachenko, K. the highest accuracy is obtained using RF respectively for multi-classification is 95.69% and for binary classification is 96.8%. The lowest value of accuracy obtained using Bernoulli NB is 55% and 72.34% respectively for binary and multi-classification. In the work by Shhadat and & Al-Sharif, the elevated value of accuracy obtained using DT is 98.2% for binary classification and for multi-classification, the elevated value of accuracy is 95.8% using RF. The lowest value of accuracy obtained using NB is 91% and 81.8% respectively for binary and multi-classification. The accuracy of the Naive Bayes classifier has increased from 55% to 91% for binary classification and from 72.34% to 81.8% for multiclass classification.

Al-Kasassbeh & Al. [4] suggest a manual feature selection method, they started by choosing randomly 100 features among 645 features. The dataset used contains 3722 Benin files from Windows XP operating system executable files and 5193 malware from Vx Heavens which contains an archive of malware datasets. These experiments consists of applying the RF model 100 times. In each time, a single feature was used to assess the importance of each feature. At the end of the experiment, only 13 features ,which give the best accuracy results, were kept. Then they implemented 6 classification modules Ridor, IBk, J48, J48 Graft, PART, RF. The results show that the RF model gives the best accuracy value close to 99% using the entire dataset, and the second experiment while keeping only 7 features, they notice that they have eliminated 6 features. This is due to the fact that although these features give a very high accuracy, they are very correlated with other features, so these 6 features have been removed. This model gives an accuracy close to 98%. To test the effectiveness of these 7 features, several comparative studies with other works they made, and each time the results of this model using this fixed number of functionalities remain relevant which means they have well done the preprocessing data phase concerning features selection.

Rathore, H., Agarwal [5] propose a malware detection model using RF and DNN, the dataset used contains 11308 malware that belong to 55 different malware families and 2819 benign files, they used Thershold method and an autoencoder for vector dimensionality reduction. The researchers implement the following models for the test: Random Forest and DNN. The first implementation of DNN is using 2 layers: 1024 and 32 nodes, then using 4 layers: 1024, 256, 64 and 16 nodes and finally using 7 layers: 1024, 512, 256, 128, 64, 32, 16 nodes. The best result is obtained using RF with Thershold method, they obtained an accuracy value equal to 99.78%.

3. DIMENSIONALITY REDUCTION

3.1 Principal Component Analysis (PCA)

PCA works in an autonomous way (unsupervised method), it allows choosing the components called Principal Components from a large amount of data in order to reduce the numbers of these components while keeping 80% (inertia) of the dataset represented by the components generated, and can be applied to build most representative features in so many fields of industry [7].

Principal components aim to study the similarity between the variables using a similarity matrix to calculate matrices allowing the projection of variables into the new space. The covariance and the Pearson correlation coefficient are two widely used methods as an index of similarity. There are extensions of the PCA like CA-PCA [8], which is used for the extraction of the features, its operation is as follows: The preprocessed data is coded, then an increase of information is applied for the classes of the dataset to maximize the effect of this information on the classes in the features extraction phase, after that a transformation matrix is chosen and the PCA is applied to the data which are augmented.

Pearson Correlation formula:

$$r_{x,y} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2} \quad (1)$$

Covariance Formula:

$$cov_{x,y} = \frac{\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})}{N - 1} \quad (2)$$

3.2 Chi-square

The chi-square is widely used in economics, biology, cryptography, and a variety of other subjects [10] such in [9]. For the study of relationships between ordinal variables. The first hypothesis of chi-square method (which we will apply on our model) is also called the zero hypothesis consists of the study of the relationship of independence between two groups (features) and to examine this hypothesis. Chi-square is used for two types of processing: independence tests and goodness-of-fit tests.

Cross tabulations, i.e. tables which allow to display the distribution relations of a variable from another variable, allows to study the relation between two variables of the same table. This method takes the form of a multidimensional table and allows to know the categorical variables which has the same characteristics. Chi-square includes 3 types of tests: independence, homogeneity and fit test; the most important of those tests is the test of independence, which makes it possible to study the relation between two variables in a population as in the following three cases: Firstly, the two variables are qualitative, secondly the two variables are quantitative and lastly the variable is qualitative and the other is quantitative. This allows us to justify the existing relations between X which is the explanatory variable and the explained variable Y with calculations of chi-square.

The chi-square method is calculated as follows :

$$X^2 = \sum_{i=1}^{n} \frac{(O_i - E_i)^2}{E_i} \tag{3}$$

Where O_i is the explanatory frequency and E_i is the explained frequency.

3.3 Recursive Feature Elimination (RFE)

RFE is an iterative procedure [11], that uses the filter-based features selection, it is an iterative method, which makes it possible to select not a single feature at a time, but a subset of features, and to delete uninformative features biasing the results [12]. Two parameters are taken into consideration for the configuration of the RFE: the number of selected features and the algorithm that will select the features. It is a recursive feature selection method, i.e. it allows to implement the machine learning model without selecting any feature, after this step we classify the features according to their importance from most important to least important[13] using a machine learning models such as DT or statistics. The least important parameters are then included in the next tests in order to reach the desired number of parameters at the end of the RFE procedure.

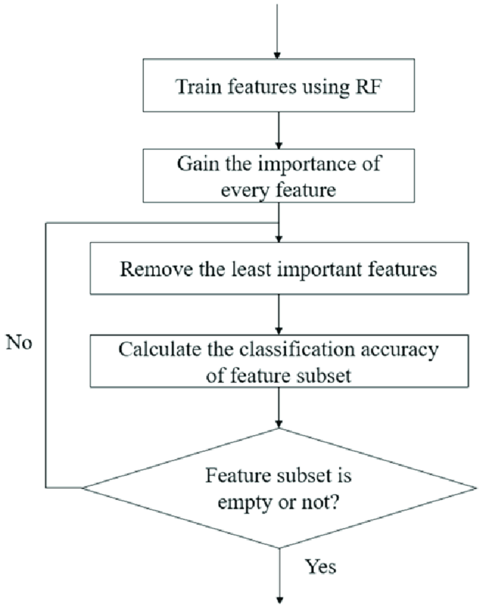
Train features using RF
Gain the importance of every feature
Remove the least important features
Calculate the classification accuracy of feature subset
Feature subset is empty or not?
No
Yes

Fig. 1. RFE Process Steps [14]

4. BUILDING THE LEARNING MODEL

4.1 Malware Detection Process

This experiment aims to implement a binary classification model, the two classes are: a class for malicious files and a class for benign files.

Figure 2 shows the five steps process that we had developed to implement the model: finding the dataset, selecting and extracting the most important and significant features, reducing the dimensionality, implementing the classification models and examining the results.

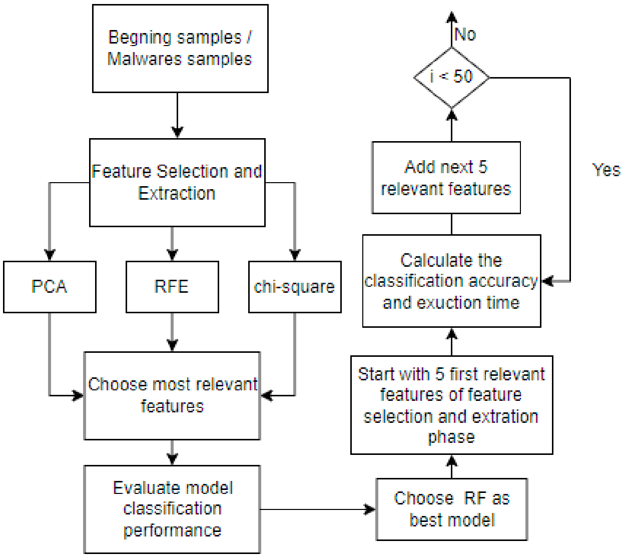
Begning samples / Malwares samples
Feature Selection and Extraction
PCA
RFE
chi-square
Choose most relevant features
Evaluate model classification performance
Choose RF as best model
Start with 5 first relevant features of feature selection and extration phase
Calculate the classification accuracy and exuction time
Add next 5 relevant features
i < 50
No
Yes

Fig. 2. Malware Detection Process

4.2 Dataset Statistical Information

The dataset on which we tested the machine learning models is obtained from Kaggle [6]. This dataset is in the form of 'PE HEADER' (Portable Executable Header), it contains 138048 lines and 57 features. The dataset is composed of two parts: a part of benign files and a part of malicious files such as Spyware, Adware, Rootkit, Backdoors, etc. For the training phase, 70% of the dataset was used for this purpose, and 30% for the test.

4.3 Implemented Models

In our study, we used many ML models to build our classifiers. We divided our work into four parts:
1- Random Forest model relies on the implementation of decision tree, as the number of decision tree increases in RF model, the better the model gives higher results. Random Forest can be trained effectively without having to normalize the features [15]. In random fores model, we used default parameters, the number of DT in RF model is equal to 100 and the maximum tree depth is not specified, so the nodes are expanded until all leaves are pure.
1-RF is a model of which we have a few case of overfitting [16], we can use the number of trees that we want, and it is a model which is very fast. Among the advantages of RF, is that it takes less time to perform. Moreover, it can classify a large amount of data with a very small error percentage .
2- Logistic Regression is an ML method for making predictions, the difference between the LR method and linear regression is that the variables to be predicted vary between zero and one. Logistic regression is a statistical and classical learning method, which includes two methods: logistic regression between two variables and logistic regression of several variables (more than two variables). Logistic Regression is divided into Binary Logistic Regression and Multivariate Logistic Regression [17]. LR is a method of statistics widely used to make the binary classification,

that is to say the classification where the explained variable Y varies between 0 and 1. The explained variable Y is calculated from one or more explanatory variables which can be categorical or ordinal variables. The activation function of the LR model avoids the use of a straight line or a hyperplane with a difference that the results of LR are between 0 and 1. The activation function used in LR is the Sigmoid function. The Logistic Regression method using other methods like Anova F-Test can be used for the detection of polymorphic malware with an accuracy of 97.7% [18].

$$y = \frac{1}{1 + e^{-value}} \tag{4}$$

The calculated values of the explained variable Y are as we have already said between the value zero and one, and underrepresented in a plane in the form of a curve that looks like an "S".

3- One of the simplest models in Machine learning is KNN,is an instance-based model[19], which has no parameters and which allows assigning a class label [20] to a new instance according to the vote of its K nearest neighbors, it is an unsupervised method that is used for classification and regression, but generally it is widely used in models for classification purposes. To avoid class conflicts, K must be chosen correctly, When the dataset is large and noisy, KNN fails to achieve the optimum of classification [21]. The principle of KNN is that it deals with the fact that the things which have a great similarity are close, similarity here means the distance between the things, the proximity between the things or even the closeness between the things, in other words, the things that are closer in the sense of distance are very similar. The implementation of the KNN consists in the choice of the parameter k, which can be fixed after several implementations of the KNN, this optimum k is chosen by relying on the model, which gives the minimum number of classification errors of the observations of the dataset. KNN is widely used in technologies of image recognition or models of decision making, etc. However, the widely used method for KNN model similarity studies is Euclidean distance. The classification is calculated from a simple majority vote of the k nearest neighbors of each point, in our model we have k = 5, so the model calculates the distance between any point of dataset and the first 5 nearest neighbors with the metric "Minkowski" and parameter p=2, which means we calculate the distance using Euclidean method.

$$d(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \tag{5}$$

4- ANN is a neural network which contains several necessarily connected layers, this network consists of three layers, an input layer, hidden layers and an output layer, each layer contains several layers, the nodes of the different layers are connected. ANN allows to study the relationships between the input variables and the output variables in order to find the links between them. The major importance of ANN networks lies in high speed data processing and massive parallel implementation, which has supported the need for research in this area [22]. To have a deep model, we just add hidden layers. ANN is the most used in the fields of image recognition, speech recognition, translation, etc. One of the most well-known networks is the Google search algorithm. The activation function used in our study is the logistic function, then we have used the 'relu' method and 4 hidden layers. 128, 16, 8 and 4 neurons were also used respectively in the layers 1, 2, 3 and 4 to give the best prediction.

5. EXPERIMENTAL RESULTS AND DISCUSSION

Table 1. models accuracy

		Feature Extraction	Feature Selection	
	All	PCA	Chi-square	RFE
ANN	99.10%	99.20%	98.70%	99.10%
RF	**99.50** %	**99.30** %	**99.40** %	**99.40**%
LR	97.20%	97.10%	95.40%	97.30%
KNN	99.00%	99.00%	98.80%	99.00%

Table 1 contains the results of the accuracy using different models. As we can see, without using a feature selection method, treatments will be done on the entire dataset without any feature selection algorithms. The best result of accuracy is obtained by the RF model (99.50%), and the lowest one by the LR method with a value of 97.20% of accuracy.

the RFE method, which allowed us to decrease the execution time of the methods as well as the complexity and increase the performances by keeping only the most significant features, we find a decrease for Random Forest model with a rate of 0.1%, an increase for LR model with a rate of 0.1% but both KNN and ANN are stable respectively at 99% and 99.10%. For the chi-square method, if we compare the results obtained with the use of the dataset in its initial state, we see that we have a decrease of accuracy for all models, an increase of 0.1%, 0.4%, 0.2% and 1.8% for respectively RF, ANN, KNN and LR.

The usefulness of the use of PCA is to transform the correlated variables into new uncorrelated variables, this allows us to reduce the execution time and algorithms complexity (we will test this hypothesis in the second experiment of this study), the increase or decrease of the results remains debatable because it depends on the dataset and the preprocessing phase. In out study, we notice that the accuracy increases using PCA for the RF and LR methods respectively with a rate of 0.2% and 0.1%, but ANN model increase with a rate of 0.1% and KNN model's accuracy remains stable at 99%.

5.1 Influence of Number of Features on Execution Time and Accuracy

According to the implementations of the models already made, we find out that the RF model gives the best overall results. In this experiment, we will test the influence of number of features change on the execution time of the model and on the accuracy.

Table 2 shows that for n = 5, the chi-square method is the best method among other feature extraction and selection methods. For PCA, we have the second best result for n = 5. From n = 10 to n = 30, we have an increase of 0.30%, after that, the accuracy remains constant. From n = 30, the results remain constant. For chi-square, after n = 10, it

Table 2. Behavior of the RF model on dataset in terms of the accuracy

		Feature Extraction	Feature Selection	
	All Features	PCA	Chi-square	RFE
n=5	99.51 %	99.00%	98.18%	97.23%
n=10	99.51 %	99.22 %	99.42 %	98.97 %
n=15	99.51 %	99.25%	99.44%	98.99%
n=20	99.51 %	99.29%	99.48%	99.15%
n=25	99.51 %	99.32%	99.49%	99.23%
n=30	99.51 %	99.33%	99.48%	99.31%
n=35	99.51 %	99.32%	99.51%	99.42%
n=40	99.51 %	99.33%	99.48%	99.42%
n=45	99.51 %	99.34%	99.50%	99.40%
n=50	99.51 %	99.33%	99.51%	99.44%

still gives the best results among the other methods (PCA and RFE), and for n = (35, 45 and 50) the score reached - 99.50% and 99.51% - is equal to the accuracy reached using all features (57 features). Concerning RFE, for n = 5, we have the smallest result for RF model, afterwards the results increase. Chi-square is found to give the best results among all other models. To conclude, we can say that the increase in features means the increase in methods accuracy for: PCA, chi-square and RFE.

Table 3. Evolution of execution time in terms of number of features for the RF model

		Feature Extraction	Feature Selection	
	All Features	PCA	Chi-square	RFE
n=5	25.43	18.48	7.88	7.18
n=10	25.43	29.64	10.75	9.87
n=15	25.43	29.61	11.67	10.25
n=20	25.43	43.10	15.04	13.59
n=25	25.43	57.33	18.12	17.05
n=30	25.43	56.42	18.01	17.06
n=35	25.43	55.71	17.99	16.65
n=40	25.43	68.80	18.43	19.85
n=45	25.43	69.73	18.53	19.86
n=50	25.43	79.72	21.95	21.55

The table 3 shows the execution times in seconds of the RF, depending on the change of the number of features. In normal state (using all features), the execution time of the RF model is 25.43 s. For the Chi-square, tree-based and RFE methods we have a lower execution time than the normal case, only the PCA exceeds 25.43 s. The increase of features means the increase of the execution time for two models (RFE and chi-square) which leads to reduce the difference between the execution time of these experiments and the normal state. By comparing RFE and chi-square, we notice that chi-square requires more execution time than RFE method. Among the 4 methods, we notice that from n = 25 to n = 50, the execution time of the PCA method exceeds the execution time of the normal state - using all features- by more than 3 times.

The extraction and selection methods did not have a big influence on the results in terms of accuracy increase, but they allowed us to reduce the time of model execution and models complexity, which is a very important aspect, because if we implement this model on an IPS or an antivirus, the anomaly detection time is an important factor in these kinds of systems. We conclude that the reduction of features cannot always lead to improve the desired results

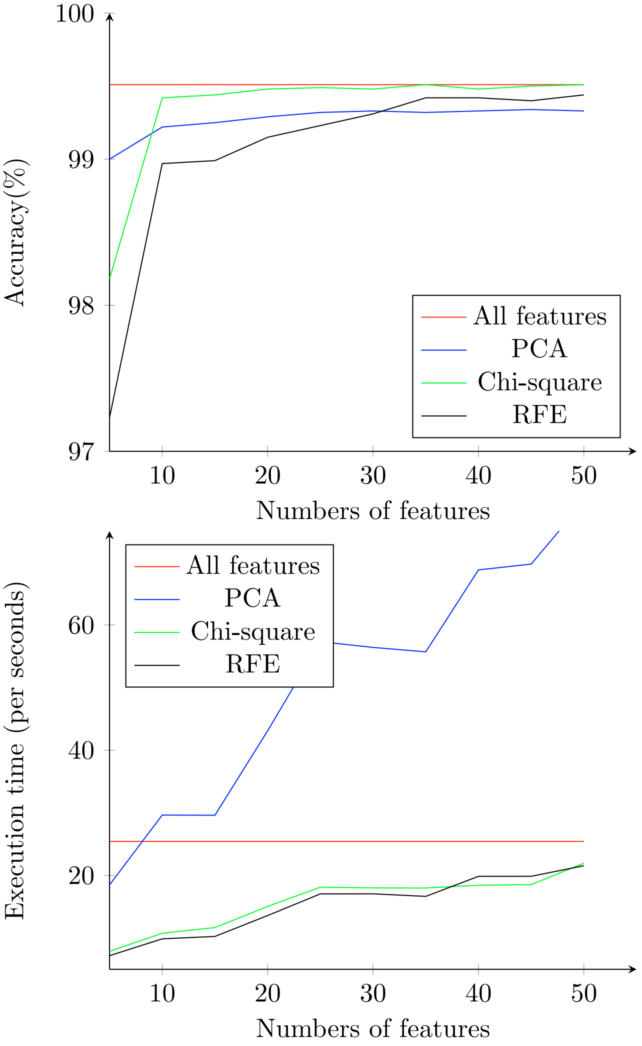

Fig. 3. Results visualization of the Random Forest model

6. CONCLUSION AND FUTURE WORK

The phase of reducing the features can influence the results either positively or negatively; in other words, there is a problem that needs to be solved: if we decrease the number of features, the quality of the results may decrease, but the process will be very fast. On the other hand, if we keep the dataset without modifications, there is a risk of falling into a state of overfitting with a very slow process. So the preprocessing phase, if not properly handled, can radically change the results and influence negatively the accuracy of the models.

The results obtained, the use of the PCA did not affect much the accuracy of these results since it just increased the execution time which can be a critical problem in case PCA was used to deploy a model of malware detection in an antivirus (Antivirus or IDS). On the other hand, the two methods of features selection improved the results due to the fact they allow to reduce the dimensionality of the dataset and decrease the execution time.

REFERENCES

[1] Computer Economics, Inc. (2007): 2007 Malware Report. The Economic Impact of Viruses, Spyware, Adware, Botnets, and Other Malicious Code. Https://www.computereconomics.com/article.cfm?id=1225

[2] Shhadat, I., Bataineh, B., Hayajneh, A., & Al-Sharif, Z. A. (2020). The Use of Machine Learning Techniques to Advance the Detection and Classification of Unknown Malware. Procedia Computer Science, 170, 917-922.

[3] Chumachenko K. Machine Learning Methods for Malware Detection and Classification. Published online 2017.

[4] Al-Kasassbeh, Mouhammd, Safaa Mohammed, Mohammad Alauthman, and Ammar Almomani. “Feature Selection Using a Machine Learning to Classify a Malware.” In Handbook of Computer Networks and Cyber Security: Principles and Paradigms, edited by Brij B. Gupta, Gregorio Martinez Perez, Dharma P. Agrawal, and Deepak Gupta, 889–904. Cham: Springer International Publishing, 2020.

[5] Rathore, H., Agarwal, S., Sahay, S. K., & Sewak, M. (2018). Malware Detection using Machine Learning and Deep Learning. arXiv:1904.02441 [cs], 11297, 402 411.

[6] Dataset Source :https://www.kaggle.com/luizbarcelos/task-1-malware-exploratory/data

[7] Zahra Berradi, Mohamed Lazaar (2019), ”Integration of Principal Component Analysis and Recurrent Neural Network to Forecast the Stock Price of Casablanca Stock Exchange”, Procedia Computer Science, Volume 148, Pages 55-61, ISSN 1877-0509.

[8] Park, M. S., & Choi, J. Y. (2009). Theoretical analysis on feature extraction capability of class-augmented PCA. Pattern Recognition, 42(11), 2353 2362.

[9] Bahassine, S., Madani, A., Al-Sarem, M., & Kissi, M. (2020). Feature selection using an improved Chi-square for Arabic text classification. Journal of King Saud University - Computer and Information Sciences, 32(2), 225 231.

[10] Ryabko, B. Ya., Stognienko, V. S., & Shokin, Yu. I. (2004). A new test for randomness and its application to some cryptographic problems. Journal of Statistical Planning and Inference, 123(2), 365 376.

[11] Guyton, F. (s. d.). Feature Selection on Permissions, Intents and APIs for Android Malware Detection. 197.

[12] Bahl, A., Hellack, B., Balas, M., Dinischiotu, A., Wiemann, M., Brinkmann, J., Luch, A., Renard, B. Y., & Haase, A. (2019). Recursive feature elimination in random forest classification supports nanomaterial grouping. NanoImpact, 15, 100179.

[13] Y. Afoudi, M. Lazaar and M. Al Achhab (2019), ”Impact of Feature selection on content-based recommendation system,” 2019 International Conference on Wireless Technologies, Embedded and Intelligent Systems (WITS), 2019, pp. 1-6, doi: 10.1109/WITS.2019.8723706.

[14] Chen, Qi, Zhaopeng Meng, Xinyi Liu, Qianguo Jin, and Ran Su. “Decision Variants for the Automatic Determination of Optimal Feature Subset in RF-RFE.” Genes 9, no. 6 (June 15, 2018): E301.

[15] Takase, H., Kobayashi, R., Kato, M., & Ohmura, R. (2020). A prototype implementation and evaluation of the malware detection mechanism for IoT devices using the processor information. International Journal of Information Security, 19(1), 71 81.

[16] Ahmad, I., Basheri, M., Iqbal, M. J., & Rahim, A. (2018). Performance Comparison of Support Vector Machine, Random Forest, and Extreme Learning Machine for Intrusion Detection. IEEE Access, 6, 33789 33795.

[17] Suhuan, L., & Xiaojun, H. (2019). Android Malware Detection Based on Logistic Regression and XGBoost. 2019 IEEE 10th International Conference on Software Engineering and Service Science (ICSESS), 528 532.

[18] Kumar, B. J., Naveen, H., Kumar, B. P., Sharma, S. S., & Villegas, J. (2017). Logistic regression for polymorphic malware detection using ANOVA F-test. 2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS), 1 5.

[19] El Mrabti S., Al Achhab M., Lazaar M. (2018), ”Comparison of Feature Selection Methods for Sentiment Analysis”. In: Lazaar M., Al Achhab M. (eds) Big Data, Cloud and Applications. BDCA 2018. Communications in Computer and Information Science, vol 872. Springer, Cham.

[20] Keller, J. M., Gray, M. R., & Givens, J. A. (1985). A fuzzy K-nearest neighbor algorithm. IEEE Transactions on Systems, Man, and Cybernetics, SMC-15(4), 580 585.

[21] Moorthy, R. S., & Pabitha, P. (2020). Optimal Detection of Phising Attack using SCA based K-NN. Procedia Computer Science, 171, 1716 1725.

[22] Izeboudjen, N., Larbes, C., & Farah, A. (2014). A new classification approach for neural networks hardware: From standards chips to embedded systems on chip. Artificial Intelligence Review, 41(4), 491 534.