TELKOMNIKA Telecommunication Computing Electronics and Control
Vol. 20, No. 1, February 2022, pp. 43~51
ISSN: 1693-6930, DOI: 10.12928/TELKOMNIKA.v20i1.18812

Comparative analysis of various machine learning algorithms for ransomware detection

Ban Mohammed Khammas
Department of Computer Networks Engineering, College of Information Engineering, Al-Nahrain University, Baghdad, Iraq

Article Info

Article history:

Received Aug 27, 2020
Revised Dec 04, 2021
Accepted Dec 13, 2021

Keywords:

Computer security
Machine learning
Ransomware detection

ABSTRACT

Recently, the ransomware attack posed a serious threat that targets a wide range of organizations and individuals for financial gain. So, there is a real need to initiate more innovative methods that are capable of proactively detect and prevent this type of attack. Multiple approaches were innovated to detect attacks using different techniques. One of these techniques is machine learning techniques which provide reasonable results, in most attack detection systems. In the current article, different machine learning techniques are tested to analyze its ability in a detection ransomware attack. The top 1000 features extracted from raw byte with the use of gain ratio as a feature selection method. Three different classifiers (decision tree (J48), random forest, radial basis function (RBF) network) available in Waikato Environment for Knowledge Analysis (WEKA) based machine learning tool are evaluated to achieve significant detection accuracy of ransomware. The result shows that random forest gave the best detection accuracy almost around 98%.

Corresponding Author:

Ban Mohammed Khammas
Department of Computer Networks Engineering, College of Information Engineering, Al-Nahrain University
Baghdad, Iraq
Email: bankhammas@coie-nahrain.edu.iq

1. INTRODUCTION

Ransomware is a type of malicious software that blocks users from accessing their device or personal data and requests ransom payment to gain access to their device. Since the first appearance of this kind in late of the 1980s till now, the ransomware witnessed a serious development that enabled the hackers to move from the personal blackmail to a high level of corporate blackmail. Therefore detecting this type of attack is a difficult technical problem [1]. The estimated cost of ransomware damage for 2017 was estimated at $5 billion, and 2019 is expected to hit $11.5 billion [2]. The Herjavec Group estimated that cybercrime will cost USD 6 trillion by 2021 [3]. In addition to major financial losses, since 2017 the risk of victimization of ransomware has risen by 97 percent [4] and the trend continues, reported that by the end of 2019 ransomware will strike a company every 14 s dropping to 11s by 2021.

In the current paper, static analysis to detect ransomware attack by extracting features directly from binary files of 32 bits size in the reprocessing stage. A gain ratio feature selection method has been used to select the best features that can be used to distinguish between ransomware and goodware samples. Besides, three different classification models have been used namely; (decision tree (J48), random forest (RF), radial basis function network (RBF)) which used the supervised learning algorithms.

The classification models are trained using 50 percent of collected ransomware files and goodware files, while the other 50 percent group is used for testing the models. The results revealed that random forest classifier is more effective in term of accuracy and time consuming compared to other classification models.The remaining parts of the article are organized as follows: section 2 addresses the related work in ransomware detection. Section 3 thoroughly discusses the proposed approach and describes the pre-processing, features extraction, feature selection, and machine learning classifiers in a comprehensive manner. Section 4 presents the dataset collection. Section 5 describes the simulation performance as well as the experimental results with the description of the evaluation studies. Finally, section 6 comprises the concluding remarks.

2. RELATED WORKS

The network security was focused attention by researchers since attacks on the computer networks became as major threats to different sectors including single user, corporate, and governmental institutions [5]. One of the most dangerous attacks is ransomware where the attacker encrypts and locks the victim's files or systems and then claims a payment to unlock and decrypt files. Many researchers studied different techniques to detect a ransomware attack.

Kharaz *et al.* [6] introduced a dynamic analysis system named UNVEIL (univeil). This technique monitors filesystem input/output (I/O) activity using the Windows filesystem mini-filter driver framework. They revealed that the system has the ability to distinguish the behavior of ransomware such as malicious encryption of files. Besides, they showed that the proposed technique could detect 13,637 ransomware samples with zero false positives from various families. Sgandurra *et al.* [7] presented a detection technique named EldeRan which is a dynamically based analysis using Sandbox. This technique monitors a set of relevant features in the first 30 seconds of the ransomware execution time. Mutual Information criterion was used as a feature selection method to select the most discriminating features. Furthermore, they utilized the regularized logistic regression classifier for the classification process. Their result achieved an area under the curve around 0.995, but at the same time, the result has a relatively high false positives ratio. Weckstén *et al.* [8] used the file system activity, registry manipulation, software process monitor, and regshots for tracking the processing activity in zeltzers. They found that the crypto-ransomware attacks depend on the executable file of "vssadmin.exe".

Vinayakumar *et al.* [9] built a system that collects the application programming interface (API) sequences from a sandbox to implement the dynamic analysis. Seven ransomware families have been used in the experiments. They employed machine learning technique represented by multilayer perceptron (MLP) for the classification process. The outcomes achieved a detection accuracy of around 98%. Chen *et al.* [10] designed a generative adversarial network (GAN) that can automatically extract dynamic features of ransomware samples. They utilized these features in different classifiers such as; (extreme gradient boosting (XGB), linear discriminant analysis (LDA), random forest, naïve Bayes, and support vector machine (SVM). The results attain an accuracy of 99%. Takeuchi *et al.* [11] applied dynamic analysis to detect ransomware by looking at the API call history run in Sandbox. They extracted API calls as features of ransomware and used SVM to classify the dataset which contains 312 goodware and 276 ransomware files. The experiments manifested an accuracy is approximately 97.48%.

Al-rimy *et al.* [12] established an ensemble-based detection model to crypto-ransomware. They combine between semi-random subspace selection (ESRS) and incremental bagging (iBagging). They compared their results with many classifiers including AdaBoost, RF, decision tree (DT), linear regression (LR), k-nearest neighbors (kNN), and SVM. The results showed an accuracy of around 0.97 when 20 features have been used in the proposed system. Homayoun *et al.* [13] combined between machine learning with sequential pattern mining to find maximal sequential patterns (MSP). The dataset contains 220 goodware samples and 1624 ransomware samples. The study comprised four classifiers namely, J48, random forest, bagging, and MLP. Their findings achieved about 99% accuracy.

Alhawi *et al.* [14] suggested a machine learning analysis model called a NetConverse. They extracted features from ransomware samples traffic. Besides, they used six types of machine learning classifiers by Waikato Environment for Knowledge Analysis (WEKA) machine learning tool. They utilized 210 samples from 9 ransomware families and 264 samples for goodware. They found that the decision tree (J48) classifier could attain a true positive ratio (TPR) of around 97.1%. Baldwin and Dehghantanha [15] also employed static analysis. They used SVM machine learning technique to classify five crypto-ransomware families and goodware. They have extracted opcode features to be used in the learning process. The outcomes emphasized an accuracy of 96.5%. Zhang *et al.* [16] proposed an approach using static analysis for ransomware classification. The technique is based on the extraction of the opcode sequences to initiate the n-gram sequences from ransomware samples and calculate the term frequency-inverse document frequency (TF-IDF) to generate feature vectors. Then, five machine learning methods are used for classification purposes. The accuracy of the proposed technique showed a percentage of 91.43%. Some works use a hybrid system that combines static and dynamic analyses such as in [17]-[19] .

Shaukat and Ribeiro [17] built a system using strong trap layer and machine learning. The experiment analysis the proposed system using 74 samples from 12 cryptographic ransomware families. The best result using gradient tree boosting algorithm has been got a detection rate around 98.25%. Meanwhile, Subedi *et al.* [18] developed an analysis tool named crypt-ransomware-static (CRSTATIC) which create dynamic-link library (DLLs) libraries from input binary programs. A data-mining technique was used to generate association rules of these DLLs. Ferrante *et al*. [19] also built a hybrid system contained the static detection method and a dynamic detection method. The static approach utilized the frequency of opcodes, while the dynamic detection method utilized system call statistics, memory usage, central processing unit (CPU) usage, and network usage to detect android ransomware. The false-positive rate attained less than 4%. The motivation of the current study is to analyze the ability of machine learning to detect ransomware using features extracted directly from the binary file, and the top frequent features extracted from ransomware files have been added to the top frequent features extracted from snort malware signatures.

3. METHODOLOGY

There is a need for a new technique that can be used in advanced security equipment which can be able to detect the security threats of ransomware. This article investigates the ability of machine learning techniques to detect ransomware by comparing three different classifiers using the proposed approach. The proposed approach, as shown in Figure 1, included three major stages. The first stage comprised a preprocessing of the dataset, while the second stage involved feature selection. The third stage implicated the use of three different classifiers to detect ransomware.

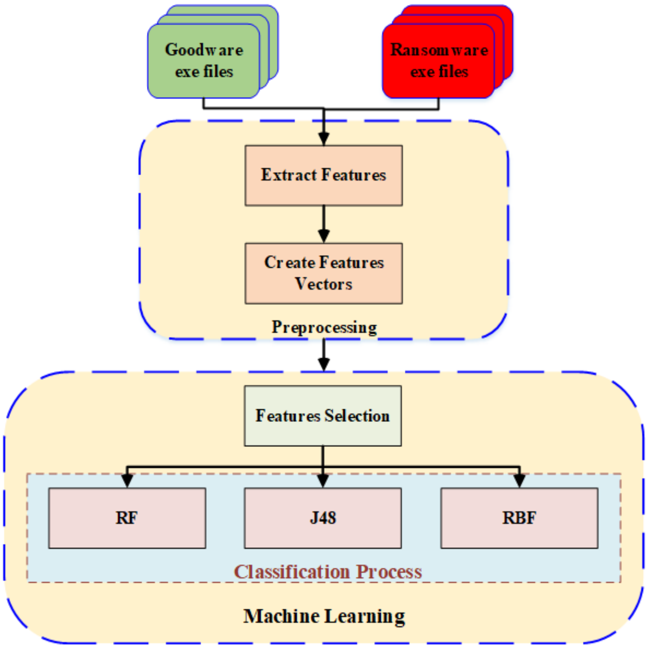


Figure 1. The framework for ransomware attack detection

In the proposed novel method, the features are extracted directly from binary files with the use of static analysis and eliminate the step of disassembling to get the opcode features. Then a preprocessing step is used to prepares the dataset and create the features vectors. This step is essentially needed because some of the symbolic features included in the raw dataset prohibiting the classifier to process this data. In the pre-processing step, the symbolic features are eliminated or changed as they do not signify crucial involvement in attack detection. Besides, these features involve undesirable effects such as increasing training time, wasted computing resources and memory, and further complexity to classifier's architecture [20].

The pre-processing step involved several sub-process. First, the raw bytes in each file is divided into a fixed-size sliding window (32-bits) in order to extract the features, since dealing with bytes is more straightforward and faster than using opcode features [21]-[23]. The feature size of 32 bit has been adopted in the current study because it produces significant results in malware detection [24]-[27]. Secondly, a counting process for the frequency of each feature in these files is implementing. According to Homayoun *et al.* [13], there are common features available in each ransomware family. Therefore, the current work focused to select these important features by analyzing each ransomware file using the counting process. The third sub-process is a normalization step which is necessary to create the feature vectors according as shown in (1).

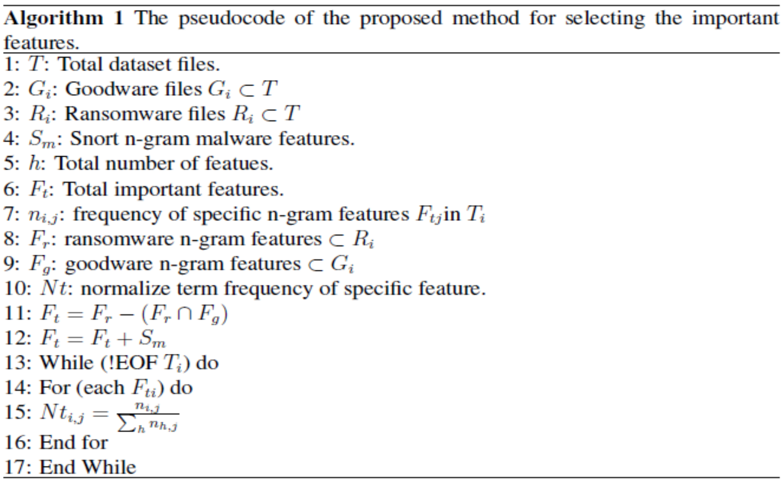
$$Nt = \frac{n_{i,j}}{\sum_h n_{h,j}} \tag{1}$$

Where Nt is the normalized frequency, $\sum_h n_{h,j}$ is the total number of features in a file, and $n_{i,j}$ is the frequency of specific features.

The second stage of the proposed method is the feature selection process which is considered an important part of the machine learning technique. It's generally used for improving the effectiveness of all the data mining algorithms and the performance of data classification [28]. The major function of feature selection is minimizing the dimensionality of features by eliminating irrelevant features. In current work, the gain ratio (GR) feature selection method has been employed where the top 1000 features are selected based on this feature selection method.

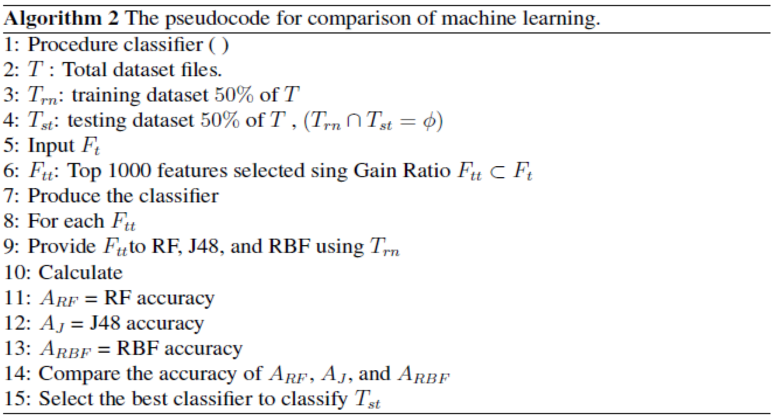
The third stage in the current proposed approach is the classification process. Three different classifiers are examined in order to find the best classifier for the detection of ransomware. These classifiers comprising decision tree (J48), random forest (RF), and radial basis functions (RBF) which have been applied using WEKA tool (an open-source graphical user interface (GUI) based machine learning tool). The decision tree is an algorithm that creates a hierarchical set of rules based on minimizing classification error developed by Quinlan [29]. The random forest algorithm is combining the results of many decision trees in order to identify the optimal set of rules that minimize the classification error. It randomly selects subsamples of features iteratively to train multiple decision trees and then built the classifier which can predict in the testing phase [30]-[32].

The radial basis functions (RBF) is a supervised learning technique that minimizing squared error. It is a neural network that has radially symmetric functional activations in the hidden layer, which means its output depends on the distance between the input data vector and the weight vector, called the center [33]. The fitness function measured is utilized to reach the best accuracy in radial basis function network (RBFN). Many fitness functions can be used to measure an error. The mean square error (MSE) has been used in current research. The pseudo-code of the proposed method which describes the procedure of selecting the important features and the pseudo-code for the comparison of the machine learning models is illustrated as shown in Algorithm 1 and Algorithm 2 respectively.



Algorithm 1 The pseudocode of the proposed method for selecting the important features.

```
$T$: Total dataset files.
$G_i$: Goodware files $G_i \subset T$
$R_i$: Ransomware files $R_i \subset T$
$S_m$: Snort n-gram malware features.
$h$: Total number of featues.
$F_t$: Total important features.
$n_{i,j}$: frequency of specific n-gram features $F_{tj}$in $T_i$
$F_r$: ransomware n-gram features $\subset R_i$
$F_g$: goodware n-gram features $\subset G_i$
$Nt$: normalize term frequency of specific feature.
$F_t = F_r - (F_r \cap F_g)$
$F_t = F_t + S_m$
While (!EOF $T_i$) do
    For (each $F_{ti}$) do
        $Nt_{i,j} = \frac{n_{i,j}}{\sum_h n_{h,j}}$
    End for
End While
```



Algorithm 2 The pseudocode for comparison of machine learning.

```
Procedure classifier ( )
$T$ : Total dataset files.
$T_{rn}$: training dataset $50\%$ of $T$
$T_{st}$: testing dataset $50\%$ of $T$ , $(T_{rn} \cap T_{st} = \phi)$
Input $F_t$
$F_{tt}$: Top 1000 features selected sing Gain Ratio $F_{tt} \subset F_t$
Produce the classifier
For each $F_{tt}$
Provide $F_{tt}$to RF, J48, and RBF using $T_{rn}$
Calculate
$A_{RF}$ = RF accuracy
$A_J$ = J48 accuracy
$A_{RBF}$ = RBF accuracy
Compare the accuracy of $A_{RF}$, $A_J$, and $A_{RBF}$
Select the best classifier to classify $T_{st}$
```

4. DATASET COLLECTION

Two types of executable files are used in the present study: ransomware executable files and goodware executable files. The ransomware files are downloaded from virustotal [34], while the goodware files are collected from the portable apps platform [35] and windows platform. The total number of ransomware files is 840 from three different families of ransomware; Cerber, Locky, and TeslaCrypt similar to [36]. The collected goodware files have almost the same size as ransomware files and the same number of 840 files. Virustotal.com has been used to check the goodware and ransomware. 50% of the dataset is used in the training stage, while the rest 50% of the dataset is used in the testing stage in order to avoid the problem of the imbalanced dataset. In the present work, two operating systems have been used to implement the proposed method and getting the results. The first one is Windows 10, Core i7 CPU with 8 core, and 16 GB of RAM. The second operating system is Linux 4.1.

5. EXPERIMENTAL RESULTS AND ANALYSIS

One of the challenges that face the researchers in the detection system is the scalability which involves; high storage requirements, more-time for implementation, and complexity. To avoid the scalability effects, different sizes of attributes are tested using GR to find the best size that offers higher accuracy in reasonable feature size. The number of 1000 attributes is found to be the best in terms of accuracy and time-consume. Figure 2 shows the simulation of the training and testing stages for the classifiers used in the proposed method.

In order to study the effectiveness of the classifiers, the false positive ratio (FPR), false negative ratio (FNR), true negative ratio (TNR), true positive ratio (TPR), and accuracy have been used in current work [36], as follows:

$$TPR\ or\ Recall = \frac{TP}{TP+FN},\ FPR = \frac{FP}{FP+TN},\ Precision = \frac{TP}{TP+FP}$$

$$TNR = \frac{TN}{TN+FP},\ Accuracy = \frac{TP+TN}{TP+FP+TN+FN},\ \mathrm{F-Measure} = 2 * \frac{(Precision*Recall)}{Precision+Recall}$$

Where:
True positive (TP): the number of attack files that are exactly predicted as attack files.
True negative (TN): the number of goodware files that are exactly classified as goodware files.
False positive (FP): the number of goodware files that are incorrectly predicted as attack files.
False negative (FN): the number of attack files that are incorrectly predicted as goodware files.

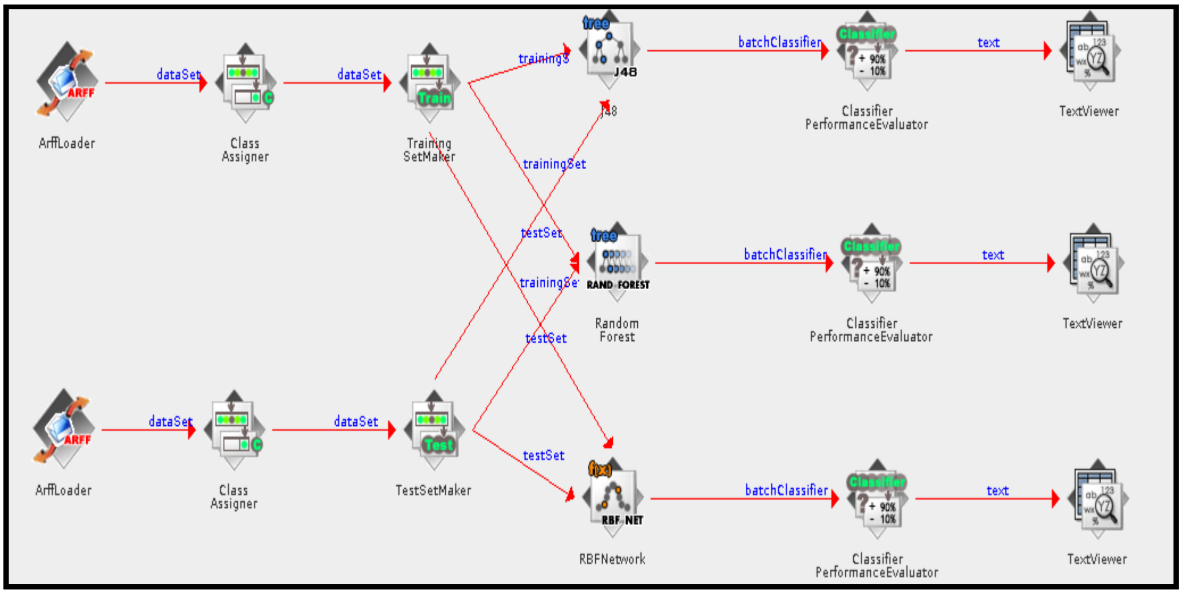


Figure 2. The simulation of the training and testing phase

To measure the accuracy of detection for different classifiers, the experiments are set to the default number for all the parameters of the different classifiers. The result of the detection accuracy using a different number of the attribute (from 1000 to 7000) is shown in Figure 3 which illustrates the best accuracy (97.73%) when using RF with 1000 attributes. Figure 4 shows the time needs for different classifiers to predict the testing dataset when the size of attributes is within the range from (1000 to 7000). The results of attributes less than (<1000) and more than (>7000) are not included in the current analysis because the detection accuracy for these ranges is very low for different classifiers. This is in line with [24] which mentioned that using a large number of attributes declines the accuracy to build the classifier model. Figure 4 depicts that the faster classifier in detection is J48 (0.54 sec.) for different sizes of attributes, while RBF shows the highest time (2.2 sec.) for a prediction than RF. Although the RF time prediction (1.49 sec.) is not the lowest, its highest accuracy makes it prevalent over other classifications. Figures 5, 6, 7, and 8 demonstrate the trends of the recall, the precision, f-measure, and receiver oprating characteristic (ROC) respectively, of the different classifiers using the different number of attributes.

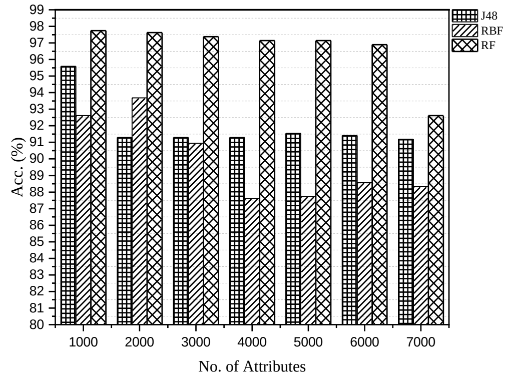


Figure 3. The accuracy of different classifiers using different sizes of attributes

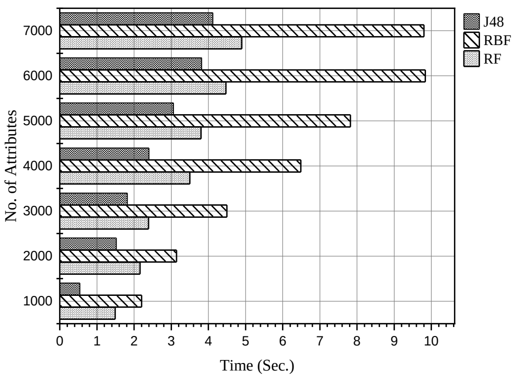


Figure 4. Time of different classifier to predict the testing dataset using different number of attributes

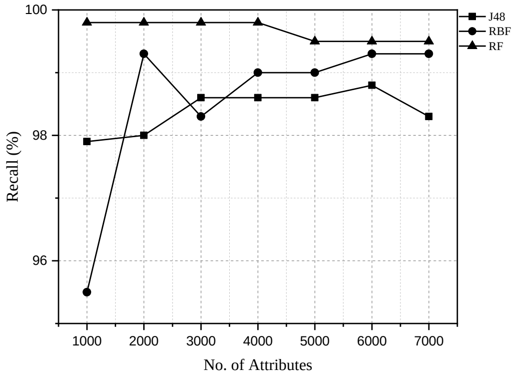


Figure 5. The recall for different classifiers with a different number of attributes

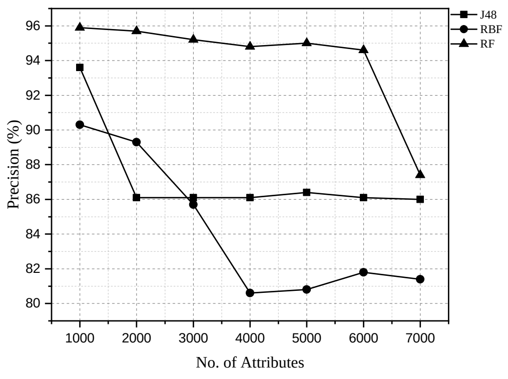


Figure 6. The precision of different classifiers with a different number of attributes

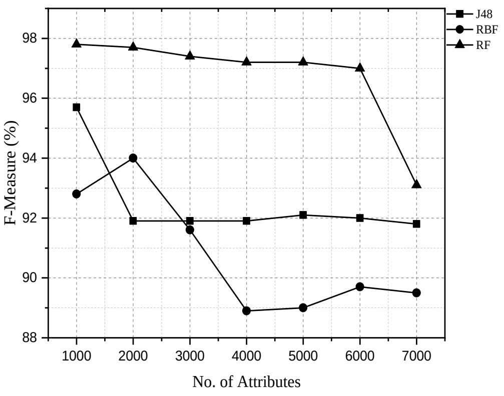


Figure 7. The F-Measure of different classifiers with a different number of attributes

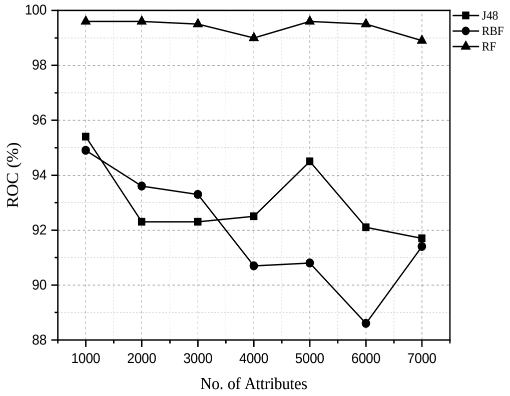


Figure 8. The ROC of different classifiers with a different number of attributes

It can be seen that the random forest achieved the best results for all previous parameters for different sizes of attributes as follows: (f-measure is 97.8, recall is 99.8, ROC is 99.6, precision is 95.9). At the same time, the result of the random forest shows that when the number of attributes increases then values of the recall, precision, f-measure, and ROC will be decreased. This finding shows that the number of attributes has a significant effect on the classifier accuracy because some of the irrelevant attributes or features in data can decrease the accuracy [24].

The FNR, FPR, and TNR are shown in Figures 9, 10, 11 respectively. As it is evident, the random forest has the highest TNR (0.957), the lowest FPR (0.043), and the lowest FNR (0.002). To compare the present work with other previous researches, Table 1 shows a comparison with the most related works. It can be seen a privilege of the proposed method over the other methods of [24] and [16].

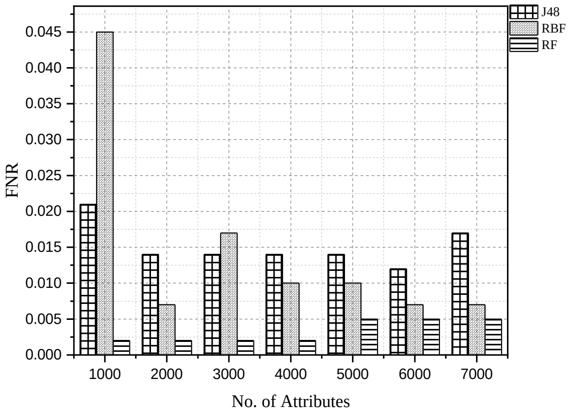


Figure 9. False negative rate

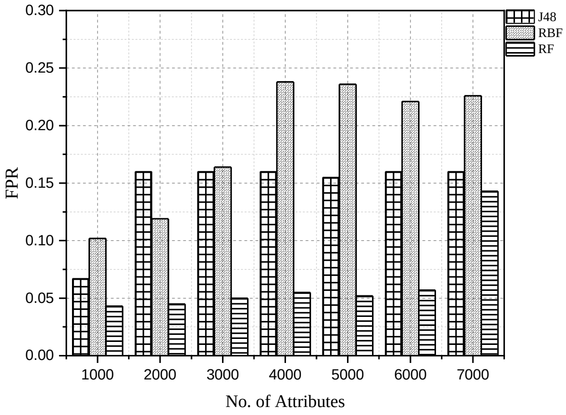


Figure 10. False positive rate

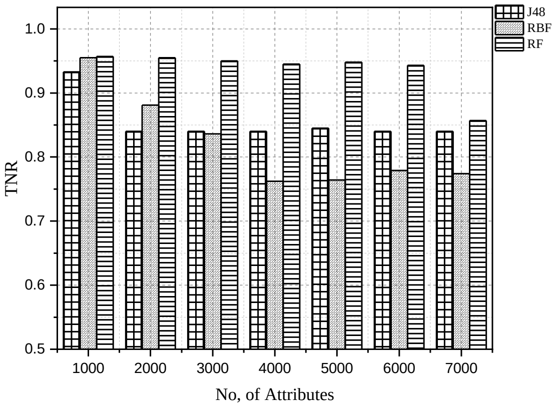


Figure 11. True negative rate

Table 1. The comparison with other related works

Method	Analyzing type	Features type	Classifier	Result accuracy
Zhang *et al.* [16]	static	Opcod (n-gram)	RF	91.4%
Baldwin and Dehghantanha [15]	static	Opcod	SVM	96.5%
Present work	static	Binary (n-gram)	RF	97.7%

6. CONCLUSION

Present work aimed to utilize the ability of machine learning techniques in a detection ransomware attack. The importance of this paper relies on using the features extracted directly from the raw byte of the executable file with the use of machine learning techniques. Three classification algorithms have been utilized in the current study including random forest, J48, and radial basis functions network. Its found that random forest is most precise in detection ransomware using the proposed method. The most suitable size was found to be 1000 attributes in the feature selection process.

The results illustrated that the random forest achieved the best results of all the measured parameters for different sizes of attributes as follows: (f-measure is 97.8, recall is 99.8, ROC is 99.6, and precision is 95.9). At the same time, these results revealed that when the number of attributes increases then the values of the recall, precision, f-measure, and ROC will be decreased. This finding referred that the number of attributes has a significant effect on the classifier accuracy because some of the irrelevant attributes or features in data can decrease the accuracy. The privilege of the proposed method is manifested in the direct extraction of features from binary files without the need of using opcode features which takes more time in the reprocessing stage due to the disassemble process.

REFERENCES

[1] D. F. Sittig and H. Singh, "A socio-technical approach to preventing, mitigating, and recovering from ransomware attacks," *Applied clinical informatics,* vol. 7, no. 2, pp. 624-632, 2016, doi: 10.4338/ACI-2016-04-SOA-0064.

[2] S. Morgan, "Global ransomware damage costs predicted to reach $20 billion (USD) by 2021," *Cybercrime Magazine,* 2019. [Online]. Available: https://cybersecurityventures.com/global-ransomware-damage-costs-predicted-to-reach-20-billion-usd-by-2021/

[3] S. Morgan, "Official annual cybercrime report," *Sausalito: Cybersecurity Ventures,* 2019. [Online]. Available: https://www.herjavecgroup.com/wp-content/uploads/2018/12/CV-HG-2019-Official-Annual-Cybercrime-Report.pdf

[4] B. Dobran, "27 terrifying ransomware statistics and facts you need to read," ed: PhoenixNap, 2019. [Online]. Available at: https://phoenixnap.com/blog/ransomware-statistics-facts

[5] 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, Part 3, pp. 987-994, 2018, doi: 10.1016/j.future.2017.01.019.

[6] A. Kharaz, S. Arshad, C. Mulliner, W. Robertson, and E. Kirda, "{UNVEIL}: A large-scale, automated approach to detecting ransomware," in *25th {USENIX} Security Symposium ({USENIX} Security 16),* pp. 757-772, 2016, doi: 10.1109/SANER.2017.7884603.

[7] D. Sgandurra, L. Muñoz-González, R. Mohsen, and E. C. Lupu, "Automated dynamic analysis of ransomware: Benefits, limitations and use for detection," *arXiv preprint arXiv:1609.03020,* 2016, doi: arXiv:1609.03020v1.

[8] M. Weckstén, J. Frick, A. Sjöström, and E. Järpe, "A novel method for recovery from Crypto Ransomware infections," in *2016 2nd IEEE International Conference on Computer and Communications (ICCC),* 2016, pp. 1354-1358, doi: 10.1109/CompComm.2016.7924925.

[9] R. Vinayakumar, K. Soman, K. S. Velan, and S. Ganorkar, "Evaluating shallow and deep networks for ransomware detection and classification," in *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI),* 2017, pp. 259-265, doi: 10.1109/ICACCI.2017.8125850.

[10] L. Chen, C.-Y. Yang, A. Paul, and R. Sahita, "Towards resilient machine learning for ransomware detection," *arXiv preprint arXiv:1812.09400,* 2018, doi: arXiv:1812.09400.

[11] Y. Takeuchi, K. Sakai, and S. Fukumoto, "Detecting ransomware using support vector machines," in *Proceedings of the 47th International Conference on Parallel Processing Companion,* 2018, pp. 1-6, doi: 10.1145/3229710.3229726.

[12] B. A. S. Al-rimy, M. A. Maarof, and S. Z. M. Shaid, "Crypto-ransomware early detection model using novel incremental bagging with enhanced semi-random subspace selection," *Future Generation Computer Systems,* vol. 101, pp. 476-491, 2019, doi: 10.1016/j.future.2019.06.005.

[13] S. Homayoun, A. Dehghantanha, M. Ahmadzadeh, S. Hashemi, and R. Khayami, "Know abnormal, find evil: frequent pattern mining for ransomware threat hunting and intelligence," *IEEE transactions on emerging topics in computing,* vol. 8, no. 2, pp. 341-351, 2017, doi: 10.1109/TETC.2017.2756908.

[14] O. M. Alhawi, J. Baldwin, and A. Dehghantanha, "Leveraging machine learning techniques for windows ransomware network traffic detection," in *Cyber threat intelligence,* ed: Springer, pp. 93-106, 2018, doi: 10.1007/978-3-319-73951-9_5.

[15] J. Baldwin and A. Dehghantanha, "Leveraging support vector machine for opcode density based detection of crypto-ransomware," in *Cyber threat intelligence,* ed: Springer, pp. 107-136, 2018, doi: 10.1007/978-3-319-73951-9_6.

[16] H. Zhang, X. Xiao, F. Mercaldo, S. Ni, F. Martinelli, and A. K. Sangaiah, "Classification of ransomware families with machine learning based on N-gram of opcodes," *Future Generation Computer Systems,* vol. 90, pp. 211-221, 2019, doi: 10.1016/j.future.2018.07.052.

[17] S. K. Shaukat and V. J. Ribeiro, "RansomWall: A layered defense system against cryptographic ransomware attacks using machine learning," in *2018 10th International Conference on Communication Systems & Networks (COMSNETS),* 2018, pp. 356-363, doi: 10.1109/COMSNETS.2018.8328219.

[18] K. P. Subedi, D. R. Budhathoki, and D. Dasgupta, "Forensic analysis of ransomware families using static and dynamic analysis," in *2018 IEEE Security and Privacy Workshops (SPW),* pp. 180-185, 2018, doi: 10.1109/SPW.2018.00033.

[19] A. Ferrante, M. Malek, F. Martinelli, F. Mercaldo, and J. Milosevic, "Extinguishing ransomware-a hybrid approach to android ransomware detection," in *International Symposium on Foundations and Practice of Security,* pp. 242-258, 2017, doi: 10.1007/978-3-319-75650-9_16.

[20] I. Ahmad, M. Basheri, M. J. Iqbal, and A. Rahim, "Performance comparison of support vector machine, random forest, and extreme learning machine for intrusion detection," *IEEE access,* vol. 6, pp. 33789-33795, 2018, doi: 10.1109/ACCESS.2018.2841987.

[21] I. Santos, Y. K. Penya, J. Devesa, and P. G. Bringas, "N-grams-based File Signatures for Malware Detection," *ICEIS,* vol. 9, pp. 317-320, 2009, doi: 10.5220/0001863603170320.

[22] M. G. Schultz, E. Eskin, F. Zadok, and S. J. Stolfo, "Data mining methods for detection of new malicious executables," in *Proceedings 2001 IEEE Symposium on Security and Privacy. S&P 2001,* 2000, pp. 38-49, doi: 10.1109/SECPRI.2001.924286.

[23] B. M. Khammas, S. Hasan, N. Nateq, J. S. Bassi, I. Ismail, and M. N. Marsono, "First Line Defense Against Spreading New Malware in the Network," in *2018 10th Computer Science and Electronic Engineering (CEEC),* pp. 113-118, 2018, doi: 10.1109/CEEC.2018.8674214.

[24] B. M. Khammas, A. Monemi, I. Ismail, S. M. Nor, and M. Marsono, "Metamorphic malware detection based on support vector machine classification of malware sub-signatures," *TELKOMNIKA Telecommunication Computing Electronics and Control,* vol. 14, no. 3, pp. 1157-1165, 2016, doi: 10.12928/telkomnika.v14i3.3850.

[25] B. M. Khammas, A. Monemi, J. S. Bassi, I. Ismail, S. M. Nor, and M. N. Marsono, "Feature selection and machine learning classification for malware detection," *Jurnal Teknologi*, vol. 77, no. 1, 2015, doi: 10.11113/jt.v77.3558.

[26] B. M. Khammas, I. Ismail, and M. Marsono, "Pre-filters in-transit malware packets detection in the network," *TELKOMNIKA Telecommunication Computing Electronics and Control*, vol. 17, no. 4, pp. 1706-1714, 2019, doi: 10.12928/TELKOMNIKA.v17i4.12065

[27] I. Ismail, M. N. Marsono, B. M. Khammas, and S. M. Nor, "Incorporating known malware signatures to classify new malware variants in network traffic," *International Journal of Network Management*, vol. 25, no. 6, pp. 471-489, 2015, doi: 10.1002/nem.1913.

[28] U. Ravale, N. Marathe, and P. Padiya, "Feature selection based hybrid anomaly intrusion detection system using K means and RBF kernel function," *Procedia Computer Science*, vol. 45, pp. 428-435, 2015, doi: 10.1016/j.procs.2015.03.174.

[29] J. R. Quinlan, *C4. 5: programs for machine learning*: Elsevier, 2014, .

[30] P. Burnap, R. French, F. Turner, and K. Jones, "Malware classification using self organising feature maps and machine activity data," *computers & security*, vol. 73, pp. 399-410, 2018, doi: 10.1016/j.cose.2017.11.016.

[31] G. Apruzzese, M. Colajanni, L. Ferretti, A. Guido, and M. Marchetti, "On the effectiveness of machine and deep learning for cyber security," in *2018 10th international conference on cyber Conflict (CyCon)*, 2018, pp. 371-390, doi: 10.23919/CYCON.2018.8405026.

[32] P. A. A. Resende and A. C. Drummond, "A survey of random forest based methods for intrusion detection systems," *ACM Computing Surveys (CSUR)*, vol. 51, no. 3, pp. 1-36, 2018, doi: 10.1145/3178582.

[33] M. Foqaha and M. Awad, "Hybrid Approach to Optimize the Centers of Radial Basis Function Neural Network Using Particle Swarm Optimization," *J. Comput.*, vol. 12, pp. 396-407, 2017, doi: 10.17706/jcp.12.5.396-407.

[34] S. Sample, "VirusTotal," [Online]. Available: https://www.virustotal.com

[35] L. Rare Ideas, "Portableapps. com-portable software for usb, portable and cloud drives," ed, 2018. [Online]. Available: https://portableapps.com/

[36] H. Hashemi, A. Azmoodeh, A. Hamzeh, and S. Hashemi, "Graph embedding as a new approach for unknown malware detection," *Journal of Computer Virology and Hacking Techniques*, vol. 13, pp. 153-166, 2017, doi: 10.1007/s11416-016-0278-y.

BIOGRAPHIES OF AUTHORS

Dr. Ban Mohammed Khammas received her B.Eng in Computer Engineering from Baghdad University in 1999 and 2002. where she received the MSc from the College of Electrical and Electronic Technology from Central Technical University in 2004 to 2006. She was awarded a Ph.D. from University Teknologi Malaysia in 2017 for her work on the network level malware detection based on packet payload classification. She was worked in the department of computer engineering at Baghdad University from 2003 to 2006. She works from 2006 to present as a Lecturer at the Department of Computer Networks Engineering, Collage of Information Engineering, AL-Nahrain University. Her research interests include deep learning, machine learning, and network security. Scopus Author ID: 56151984900