 doi:10.20944/preprints202201.0454.v1

Ransomware family classification with ensemble model based on behavior analysis

Nowshin Tasnim[1], Khandaker Tayef Shahriar[1], Hamed Alqahtani[2] and Iqbal H. Sarker[1,*]

[1]Dept. of Computer Science and Engineering, Chittagong University of Engineering & Technology, Chittagong-4349, Bangladesh.
[2]College of Computer Science, King Khalid University, Abha, Saudi Arabia.
[*]Correspondence: tasnimnowshin95@gmail.com, iqbal@cuet.ac.bd

Abstract. Ransomware is one of the most dangerous types of malware, which is frequently intended to spread through a network to damage the designated client by encrypting the client's vulnerable data. Conventional signature-based ransomware detection technique falls behind because it can only detect known anomalies. When it comes to new and non-familiar ransomware traditional system unveils huge shortcomings. For detecting unknown patterns and sorts of new ransomware families, *behavior-based anomaly detection* approaches are likely to be the most efficient approach. In the wake of this alarming condition, this paper presents an *ensemble classification model* consisting of three widely used machine learning techniques that include Decision Tree (DT), Random Forest (RF) and K-nearest neighbor (KNN). To achieve the best outcome ensemble *soft voting* and *hard voting* techniques are used while classifying ransomware families based on attack attributes. Performance analysis is done by comparing our proposed ensemble models with standalone models on behavioral attributes based ransomware dataset.

Keywords: Ransomware · Behavior analysis · Cyber Security · Machine Learning · Ensemble model · Supervised classification.

1 Introduction

The internet usage pattern has changed dramatically over the years. With the advent of web services, cyber threats are also increasing gradually [14]. A cyber threat is a form of malicious activity that attempts to disrupt client data which includes malware, data breaches, zero-day attacks, identity theft, and other harmful practices [19]. In recent years, ransomware has become one of the most serious digital crimes which affect organizations pitifully. The Federal Bureau of Investigation (FBI) reports that the losses incurred in 2016 due to ransomware are approximately 1 billion US dollars [4]. Brewer et al. [4] informed that in 1989, Dr. Joseph Popp circulated a trojan called PC Cyborg where a malware program would cover all the folders and clash with records on Computer's C: drive. Then the attacker sent a ransom demanding 189 US dollars to recover

the data from the affected computer and manage the malicious programs. In short, ransomware is a self-spreading malicious program that uses encryption and locking mechanisms to capture victims' information and demand a ransom for recovery. Thus, it is a great challenge to ensure the security and safety of digital documents from ransomware. To meet the challenges of the Fourth Industrial Revolution (Industry 4.0), Sarker et al. [17] provided a direction with ANN (Artificial Neural Network) and DL (Deep Learning) methods that can also be used to protect computer networks. However, in this paper, we focus on machine learning techniques that can detect the cyber-anomalies effectively [16]. Bendovschi et al. [3] reports that ransomware has exceeded 33 percent by 2020 than 2019. Hence, with the increase of ransomware attacks, it is essential to detect attacks effectively and minimize financial loss.

With the advancement of information technology, cybercriminals develop new types of attacks and tactics to make the computer system vulnerable and remain untraceable. Signature-based anomaly detection is not useful to detect ransomware because attackers could change and increase the malicious program to bypass the detection mechanisms of anti-virus softwares [12]. However, Kruegel et al. [10] presented the complexity of storing a large number of signatures of known anomalies in the signature-based ransomware detection system. The authors also suggested a strategy that evaluates each web application and compares it with standard log files to detect anomalies. Based on the concept of AI-based cybersecurity, Sarker et al. [19] presented a summary by addressing the challenges of traditional methods and focusing on data-driven intelligent decision support to protect the system from cyber-attacks on the perspective of machine learning.

In this paper, we propose a behavior-based ransomware family classification system. We collect a ransomware dataset having 85 behavioral features for the classification analysis. We select the 20 best features from 85 features by using an effective feature selection technique. We consider the correlation value less than 0.95 to avoid multicollinearity problems. The main advantage of this method is that it does not consider any traditional static methods but focuses on the dynamic prediction method. The primary contributions of this paper are given below:

- Our proposed approach performs a behavior-based analysis by using the machine learning approach and acquires the ideal precision of ransomware classification.
- The proposed approach effectively detects ransomware families by applying an ensemble voting classifier with the implementation of three machine learning techniques.
- The range of experiments presents a comparison of standalone models with the ensemble voting-based models with an evaluation of standard deviation and means of accuracy to show the effectiveness of our proposed approach.

The rest of the paper is organized as follows. Section 2 reviews the related works, section 3 precisely describes the working procedure of the proposed

 doi:10.20944/preprints202201.0454.v1

method. Section 4 contains the result and performance analysis. And finally, in section 5 we conclude the paper by summarizing the work.

2 Related Work

Ransomware is a detrimental kind of malware that can lock the victims' screen or illegally encrypt their confidential documents for ransom, resulting in significant damage to clients. Zhang et al. [20] separated ransomware by families which help to distinguish the variation of the ransomware test. They achieved an accuracy rate of 91.43 percent by using an opcode sequence for each sample of ransomware and converting it to an N-gram sequence. The classification method of Pirscoveanu et al. [14] implemented a cognitive combination of features that achieves a high degree of accuracy with a typical AUC value of 0.98 for random forest classifier. Pekta et al. [13] particularly addressed that in runtime analysis of malware, file system, network, registry activities, and API calls are the most important behavioral attributes. They also used N-gram display over API calls to separate malware families. Daku et al. [7] proposed primarily two approaches: a repetitive approach to recognize behaviors for high-level classification performance, and a collective approach for highly related behaviors. Alaeiyan et al. [1] recommended another order of trigger-based malware classification by following evasive and elicited practices. Both of these practices address the specification of environmental conditions. However, evasive practices focus on self-defense while elicited practices show the benefits of malware for malignant demonstrations. Chen et al. [6] presented how to measure ransomware behavior from a secure log called a cuckoo sandbox. They considered all logs from contaminated hosts as individual records and looked at the features from the infection report by using the TF-IDF process. Galal et al. [9] discussed statistical-based, graph-based, polymorphic and metamorphic malware structure. To avoid problems with signature-based detection Canfora et al. and Bazrafshan et al. [5, 2] provided some alternative detection methods such as obfuscation strategy and heuristic technique.

By considering the above works, in this paper, we focus on the behavior of anomaly to handle the rise of cybercrime and the problems of the traditional signature-based detection system. Moreover, our feature selection technique provides better classification accuracy of more than 97 percent in 10 different ransomware families. The proposed approach is based on an ensemble model which is incorporated with three basic machine learning classifiers.

3 Methodology

In this section, we present the methodology of our proposed approach that performs the classification process on 10 different ransomware families. We dynamically select the best 20 features out of 85 features by using the correlation matrix and the value of feature importance without any user involvement. We develop an ensemble model by implementing three popular machine learning models:

KNeighbor, Decision Tree, and Random Forest Classifier [18]. Our approach automatically selects the optimized K value for the KNeighbor classifier. Finally, we present a comparison of standalone models with the ensemble models. We illustrate the whole process of the proposed ensemble model-based ransomware classification approach in Algorithm 1.

Algorithm 1: Ransomware family classification with ensemble model

Input: Dataframe df containing ransomware instances of n different families.
Result: Predict family class label of n instances.

```
ObjList = df.select_data_type(object);
for feature in ObjList do
    df[feature] = convert to numeric
end
Correlation_matrix = df.corr();
Drop upper[columns], where correlation_value> 0.95 ;
Feed df into mutual_classif_info to find importance with respect to target label;
sort importance in descending order;
if importance <threshold then
    drop less important features;
end
for i in range (1, 40) do
    error_rate = KNeighborClassifier(k = (i + 1);
end
Select k value for minimum error_rate;
Pass optimized KNN, DT and RF to Ensemble model;
Feed dataset into Ensemble model;
Compare standalone model with ensemble model by mean value and standard
  deviation of classification accuracy;
```

At first, the algorithm takes the dataset as input that contains ransomware instances. Each instance has 85 behavioral attributes. In the next step, we perform the data preprocessing for the feature selection. Then the processed data access to the feature selection phase by implementing a two-fold method. In the first fold, the less important features are removed using the information gain method. The threshold value is automatically generated according to the data pattern. The second fold of the feature selection method considers the correlation value to remove the highly correlated features. In the final phase of the proposed algorithm, the ensemble model is implemented to classify the ransomware family.

3.1 Data Preprocessing

We collect 10 different ransomware family datasets with similar attributes from the Canadian Institute for Cybersecurity [16, 8] and combine them into a single dataset. The final dataset contains 107700 instances with 85 features and 10 different class labels. Then the system observe data to check whether it fits for the

 doi:10.20944/preprints202201.0454.v1

next procedure. Few feature selection methods do not accept non-numerical attributes. Hence, data preprocessing plays a vital role in suiting the data into the adopted feature selection method. The proposed system analyzes the features and converts non-numerical features to numerical ones. We apply the information gain method and correlation matrix [11, 15] to select features for further processing.

3.2 Feature Selection

The value of the attribute can be determined significantly if the dataset is managed to contain a large number of attributes. This type of dataset is often referred to as a high-dimensional dataset. This high dimensionality comes with a number of problems, for example, this will typically create overhead to train the machine learning model and increase the complexity of the model. Figure 1 shows that the density of the feature value is high in a portion of the feature section. Thus it is important to extract important features and remove unnecessary features to get a balanced feature set to train the model. Figure 1 reflects that feature importance values are ordered in descending manner that helps to perform simple and appropriate analysis.

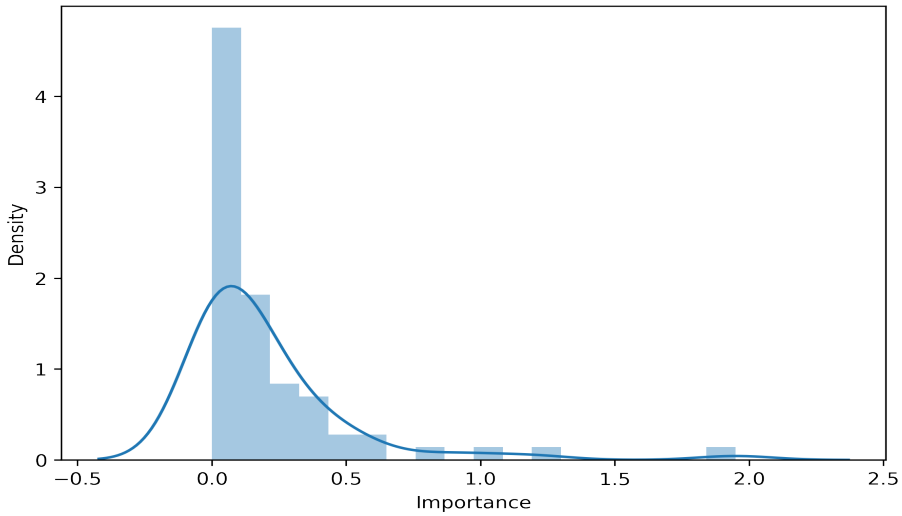

Fig. 1. Density of feature importance.

The proposed method contains two completely automated major steps to extract features dynamically. The first step is accomplished by implementing a correlation matrix. Correlation measures the internal dependency of features. The high value of the correlation is better in the sense that one attribute can be

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 January 2022 doi:10.20944/preprints202201.0454.v1

predicted from another. But often the problem arises in the case of a perfect positive or a perfect negative correlation between two attributes. Hence it is required to manage multicollinearity problems otherwise it can lead to wrong predictions. Thus the proposed system drops the attributes which have a correlation value of more than 0.95. By applying this process 17 attributes are removed and 68 attributes are extracted. In the second step of feature selection, we measure each feature's importance by concerning the target feature. Some of the features possess almost zero importance density. Fig. 2 shows the importance range with respect to the target feature. After examining all features, the Information gain method exhibits that there are a few less important features.

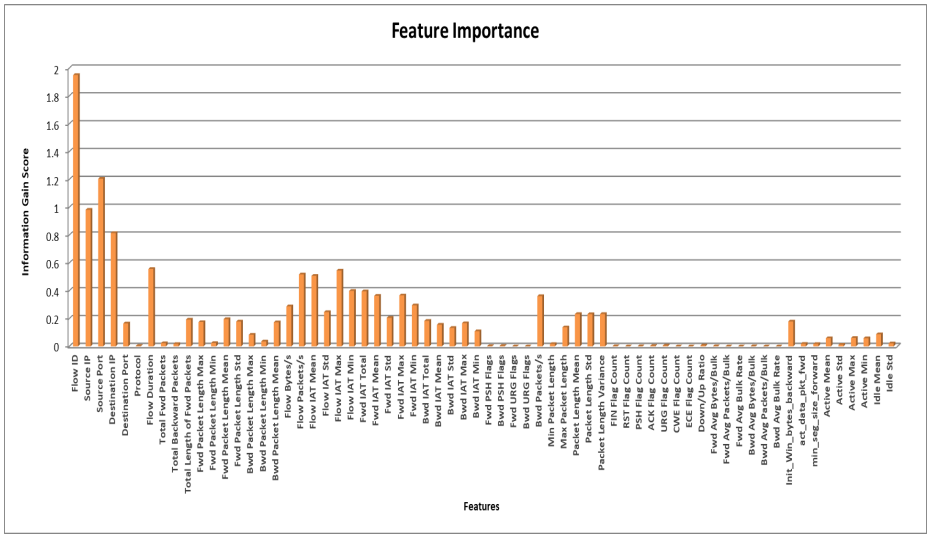

Fig. 2. Feature importance with respect to target feature.

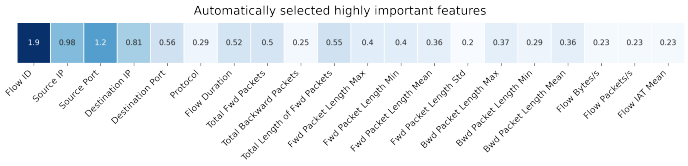

Fig. 3. Automatically selected top 20 important features.

To extract the best features we follow an automatic selection process by generating a standard threshold and selecting the most important features. The information gain method helps to identify the key attributes to classify the training

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 January 2022 doi:10.20944/preprints202201.0454.v1

data accurately. Hence, no user interpretation is required to select the number of attributes. The proposed system automatically generates necessary parameters to select features using the following equations.

$$a = pd.Series.mean(importance) \tag{1}$$

$$new\_series = importance[(importance > a).any(1)] \tag{2}$$

$$threshold = len(new\_series) \tag{3}$$

$$p = (threshold/len(importance)) * 100 \tag{4}$$

Here p is the percentile parameter to pass into the feature selection method. Our adopted technique selects p% features from the total number of attributes. However, in this process, only key attributes are selected that have a value that exceeds the mean importance value and produces a percentage of the total length of the series. The dynamic feature selection is one of the main contributions. Fig. 3 presents automatically selected features by the proposed system. Eventually, 20 features are selected. Fig. 4 shows the correlation matrix of the finally selected features, which has value less than 0.95.

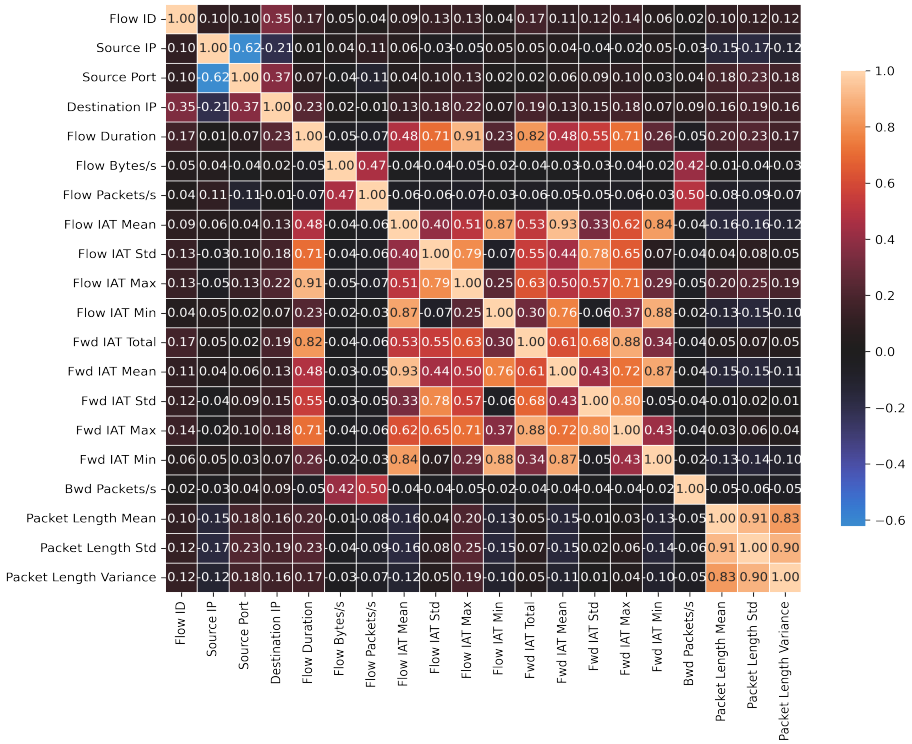

Fig. 4. Correlation matrix of finally selected features

 doi:10.20944/preprints202201.0454.v1

3.3 Ensemble Model

We propose an ensemble model to detect ransomware families effectively. The ensemble model measures the performance of multiple models with a variety of voting techniques. We analyze both hard and soft voting with a standard deviation of accuracy score. We incorporate the Decision Tree, Random Forest, and KNeighbour algorithm in the ensemble model. However, the regression models perform like predictors and show little less efficiency because the features contain high correlation. Thus, in this case, multicollinearity and high correlations between predictors often mislead the performance. Moreover, the classifiers that use the entropy measure of attributes while taking decisions get the priority because we extract features based on the information gain method. The final execution of the ensemble model is followed by selecting Decision Tree classifier, Random Forest Classifier and the optimum version of the KNeighbor classifier. In Fig. 5 error rate with respect to each k value for KNeighbor classifier is plotted. Here, the k value is 1 with a minimum error rate of -0.025. Finally, the automatically generated optimized k value is transferred to the ensemble model.

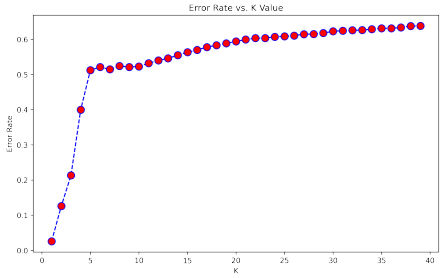

Fig. 5. Error rate w.r.t K value in KNN Classifier.

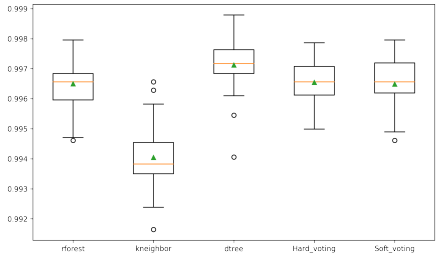

Fig. 6. Accuracy mean std. score of standalone models and ensemble voting models.

4 Experiment and Result Analysis

This section includes results analysis with a confusion matrix, precision, recall, f1-score for each ransomware family available in the dataset. The proposed ensemble model is prepared by the three machine learning classifiers. We use the mean and standard deviation methods to compare the standalone models with the ensemble models. Standard deviation is a factual estimation of the sum of numbers that fluctuate from the normal numbers in a series. A low standard deviation implies that the information is firmly associated with the normal. High standard deviation implies that there is a significant difference between the data and the mean value. Table. 1 presents that soft voting of the ensemble generates high accuracy with minimum standard deviation.

 doi:10.20944/preprints202201.0454.v1

Table 1. Comparison of Standalone Models & Ensemble Model

Model	Accuracy Mean	Accuracy Std.
DecisionTree	0.97716	0.00185
RandomForest	0.97211	0.00168
KNearestNeighbor	0.95214	0.00205
Ensemble Hard Voting	0.97299	0.00175
Ensemble Soft Voting	***0.97568***	***0.00155***

Though the accuracy mean of the Decision Tree is higher than the Soft voting of the ensemble model, but it has a greater standard deviation compared to others.

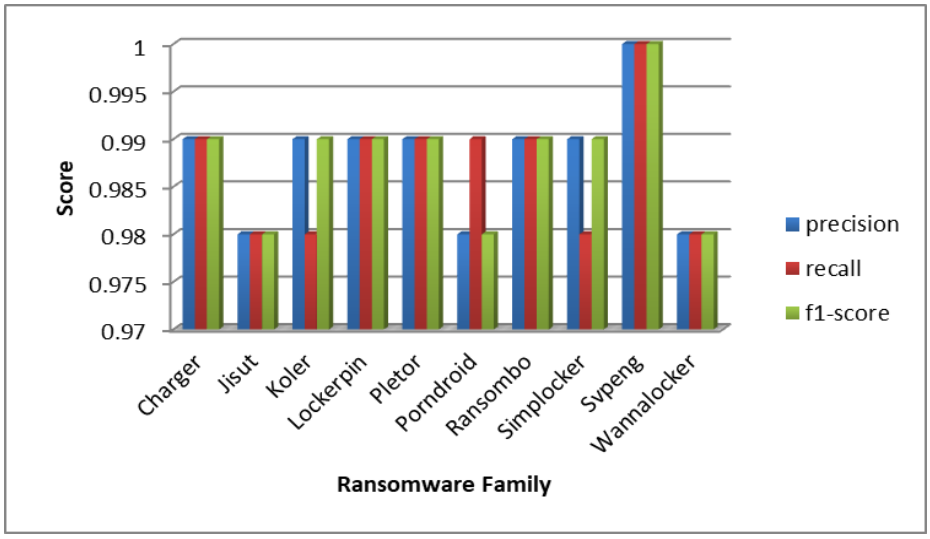

Fig. 7. Classification Report of Ensemble Soft voting model

The performance of the standalone models and ensemble models with two types of voting techniques is estimated and plotted in Fig. 6 also shows the standard deviation and mean score for each model. Though the accuracy of each model is almost similar, but according to standard deviation, soft voting for the ensemble model provides the most reliable performance by being closely related to the mean value. A large value of standard deviations is considered less reliable because it varies greatly with mean values. We present a classification report analysis of the ensemble soft voting model in Fig. 7 as it is evaluated as the best model. We use the 10-fold cross-validation method to improve the reliability of the performance of the model.

5 Conclusion

In this paper, we propose a method of classifying the ransomware family by using a unique ensemble method based on behavioral analysis. The behavior-centric detection system enables the most significant results as the attackers also work continuously to avoid security measures. Selecting the appropriate attribute to achieve the best accuracy is a great challenge. To overcome this challenge, in this paper, we compile two levels of the feature selection process by implementing the information gain method and correlation value. We also develop formulas to generate the importance threshold value automatically depending on the dataset pattern. Our proposed ensemble technique is based on Decision Tree, Random Forest, and the KNeighbor classifier where the best Kneighbor classifier version is selected with the minimum error rate. We analyze and evaluate the performance of the model by executing two types of voting classifiers: Soft voting and hard voting. Choosing the ideal standard deviation and mean value of accuracy are the important factors for the classification purpose. Classification accuracy with minimum standard deviation is considered the most reliable one. The experimental results show that the ensemble model with the soft voting classifier performs better resulting in an accuracy of 97% with a minimum standard deviation of 0.00155.

References

1. Alaeiyan, M., Parsa, S., Conti, M.: Analysis and classification of context-based malware behavior. Computer Communications **136**, 76–90 (2019)
2. Bazrafshan, Z., Hashemi, H., Fard, S.M.H., Hamzeh, A.: A survey on heuristic malware detection techniques. In: The 5th Conference on Information and Knowledge Technology. pp. 113–120. IEEE (2013)
3. Bendovschi, A.: Cyber-attacks–trends, patterns and security countermeasures. Procedia Economics and Finance **28**, 24–31 (2015)
4. Brewer, R.: Ransomware attacks: detection, prevention and cure. Network Security **2016**(9), 5–9 (2016)
5. Canfora, G., Di Sorbo, A., Mercaldo, F., Visaggio, C.A.: Obfuscation techniques against signature-based detection: a case study. In: 2015 Mobile systems technologies workshop (MST). pp. 21–26. IEEE (2015)
6. Chen, Q., Bridges, R.A.: Automated behavioral analysis of malware: A case study of wannacry ransomware. In: 2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA). pp. 454–460. IEEE (2017)
7. Daku, H., Zavarsky, P., Malik, Y.: Behavioral-based classification and identification of ransomware variants using machine learning. In: 2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE). pp. 1560–1564. IEEE (2018)
8. Ferrag, M.A., Maglaras, L., Moschoyiannis, S., Janicke, H.: Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study. Journal of Information Security and Applications **50**, 102419 (2020)

9. Galal, H.S., Mahdy, Y.B., Atiea, M.A.: Behavior-based features model for malware detection. Journal of Computer Virology and Hacking Techniques **12**(2), 59–67 (2016)
10. Kruegel, C., Vigna, G.: Anomaly detection of web-based attacks. In: Proceedings of the 10th ACM conference on Computer and communications security. pp. 251–261 (2003)
11. Lee, C., Lee, G.G.: Information gain and divergence-based feature selection for machine learning-based text categorization. Information processing & management **42**(1), 155–165 (2006)
12. Patcha, A., Park, J.M.: An overview of anomaly detection techniques: Existing solutions and latest technological trends. Computer networks **51**(12), 3448–3470 (2007)
13. Pektaş, A., Acarman, T.: Classification of malware families based on runtime behaviors. Journal of information security and applications **37**, 91–100 (2017)
14. Pirscoveanu, R.S., Hansen, S.S., Larsen, T.M., Stevanovic, M., Pedersen, J.M., Czech, A.: Analysis of malware behavior: Type classification using machine learning. In: 2015 International conference on cyber situational awareness, data analytics and assessment (CyberSA). pp. 1–7. IEEE (2015)
15. Roobaert, D., Karakoulas, G., Chawla, N.V.: Information gain, correlation and support vector machines. In: Feature extraction, pp. 463–470. Springer (2006)
16. Sarker, I.H.: Cyberlearning: Effectiveness analysis of machine learning security modeling to detect cyber-anomalies and multi-attacks. Internet of Things **14**, 100393 (2021)
17. Sarker, I.H.: Deep learning: A comprehensive overview on techniques, taxonomy, applications and research directions. SN Computer Science **2**(6), 1–20 (2021)
18. Sarker, I.H.: Machine learning: Algorithms, real-world applications and research directions. SN Computer Science **2**(3), 1–21 (2021)
19. Sarker, I.H., Furhad, M.H., Nowrozy, R.: Ai-driven cybersecurity: an overview, security intelligence modeling and research directions. SN Computer Science **2**(3), 1–18 (2021)
20. Zhang, H., Xiao, X., Mercaldo, F., Ni, S., Martinelli, F., Sangaiah, A.K.: Classification of ransomware families with machine learning based on n-gram of opcodes. Future Generation Computer Systems **90**, 211–221 (2019)