Journal of Artificial Intelligence and Data Mining (JAIDM), Vol. 12, No. 1, 2024, 95-104.

Shahrood University of Technology

Journal of Artificial Intelligence and Data Mining (JAIDM)
Journal homepage: http://jad.shahroodut.ac.ir

Research paper

A Hybrid Machine Learning Approach and Genetic Algorithm for Malware Detection

Mahdieh Maazalahi and Soodeh Hosseini*

Department of Computer Science, Faculty of Mathematics and Computer, Shahid Bahonar University of Kerman, Kerman, Iran.

Article Info

Article History:
Received 30 November 2023
Revised 24 January 2024
Accepted 09 March 2024

DOI:10.22044/jadm.2024.13895.2503

Keywords:
Malware Detection, Hybrid Method, Data Mining Algorithms, Feature Selection.

**Corresponding author: so_hosseini@uk.ac.ir (S . Hosseini).*

Abstract

Detecting and preventing malware infections in systems is become a critical necessity. This paper presents a hybrid method for malware detection, utilizing data mining algorithms such as simulated annealing (SA), support vector machine (SVM), genetic algorithm (GA), and K-means. The proposed method combines these algorithms to achieve effective malware detection. Initially, the SA-SVM method is employed for feature selection, where the SVM algorithm identifies the best features, and the SA algorithm calculates the SVM parameters. Subsequently, the GA-K-means method is utilized to identify attacks. The GA algorithm selects the best chromosome for cluster centers, and the K-means algorithm is applied to identify malware. To evaluate the performance of the proposed method, two datasets, Andro-Autopsy and CICMalDroid 2020, have been utilized. The evaluation results demonstrate that the proposed method achieves high true positive rates (0.964, 0.985), true negative rates (0.985, 0.989), low false negative rates (0.036, 0.015), and false positive rates (0.022, 0.043). This indicates that the method effectively detects malware while reasonably minimizing false identifications.

1. Introduction

Any software that can infect the network has been known as malware [1]. With the advancement of information technologies and services, the number of their users has grown, which has increased the information stored in them, making it an accessible target for attackers. There are different types of malware such as viruses, worms, Trojans, ransomware, etc. [2].

With the increasing spread of different types of malware, various methods are proposed for detection, such as 1) signature-based detection, 2) behavior-based malware detection, 3) discovery-based malware detection 4) model-based malware detection 5) Internet of Things-based malware detection 6) Mobile device-based malware detection, and 7) Cloud-based malware detection. Malware detection is based on deep learning [3]. It has determined whether the program has a malicious purpose or not, and once it has determined whether the program has a malicious purpose, it has identified the malware using analysis and detection [4].

In this paper, a hybrid method called SA-SVM-GA-K-means for malware detection is presented. The contributions are listed as follows. In this paper, a dynamic hybrid method using data mining algorithms to identify and classify malware based on network flows and maintain the security of Internet networks is developed to solve complex multi-objective problems, with high convergence speed to minimize the required costs (storage space, computation time, computational complexity), and try to find the global optimal solution by discovering new local regions. This method identifies a large set of malware families from network flows and classifies them into a specific malware type or malware family.

This method is not dependent on a specific tool and does not represent a limitation for the system. The evaluation shows that it outperforms other integrated models and is designed for large datasets and avoids overfitting.

The identification of malware in network-based datasets has a high computational complexity due to its high dimensions. In this method, by choosing the optimal features, the dimensions of the problem are reduced and the k-means algorithm is used to estimate the dynamic search space of the GA algorithm, which compared to the entire original search space, the population size and the number of generations are reduced, and the initial space is much less required, thus reducing the computational complexity.

The proposed method's suitable features are selected using the SVM algorithm. It is difficult and complex to determine the appropriate hyper SVM parameters (Cost-C and Gamma). 1) In each iteration to select the parameters, their performance has been evaluated using the entropy feature, which has been able to introduce suitable random elements, and this random value contributes a lot to prevent the SVM algorithm from falling into the local optimum, which improve the ability of the SVM algorithm in the optimum Globalization increases the accuracy of classifiers and reduces the amount of memory used. 2) Creating a separation screen using the SA algorithm has made it possible to separate different types of data such as linear and non-linear.

Malware is identified using the GA-K-Means algorithm. 1) It has increased the convergence speed. 2) In this method, non-objective reference point data is provided for the automatic determination of the number and direction of the subspace vector using the K-means algorithm. 3) In this method, due to the high competition by evaluating each chromosome with the K-NN algorithm, a list of optimal solutions could be generated and improved over time. 4) In this algorithm, a large number of chromosomes are stored in each step, which requires a lot of space. Here the K-means algorithm is used, which limits the number of chromosomes. The generated chromosomes are selected as cluster centers of the K-means algorithm, which reduces the storage cost. 5) The GA population is first initialized using the hybrid SA-SVM algorithm (determining the number of chromosomes) and to overcome the limitations of the GA algorithm, the K-means algorithm is applied to the new mutation that depends on the endpoints.

The proposed method is evaluated in a real environment consisting of both clean malware traffic and noisy traffic. The robustness of the system in real-world conditions was compared using two datasets, Andro-Autopsy and CICMalDroid 2020, which contain different types of malware, and 6 other classifications based on data mining algorithms.

The rest of the paper is organized as follows. Section 2 provides a summary of the work done on malware detection. Section 3 gives a full description of the proposed hybrid method (SA-SVM-GA-K-means). Section 4 shows the evaluation of the proposed method. Finally, in section 5, conclusions and future work are presented.

2. Related Work

This section describes the work carried out in recent years to explain the malware detection using a combination of data mining algorithms and briefly compares them in Table 1.

Alamro *et al.* [1] proposed a method called AAMD-OELAC. The proposed method includes data preprocessing, ensemble learning, and hyperparameter tuning. LS-SVM, KELM, and RRVFLN methods are used to identify malware. The proposed method has been evaluated using the Andro-Autopsy dataset. The evaluation results have shown that the proposed method has an accuracy of 0.989. Yumlembam *et al.*[2] proposed an intrusion detection method that uses the Graph Neural Networks (GNN) algorithm to select features and uses the adversarial network (GAN) to classify malware. This system has been evaluated using CICMaldroid and Drebin datasets, and the results show that the proposed method has detected malware with the lowest error rate. Kim *et al.* [3] proposed a malware detection system called MAPAS. This system is based on graphs of API calls using Convolutional Neural Networks (CNN) using the common features of the graphs, and then a lightweight classification algorithm is used to identify malware. This method has been evaluated using the MaMa Droid dataset, and the evaluation results have shown that it has detected attacks with an accuracy of 0.930. Anand *et al.* [4] proposed a new deep learning method called CNN-DMA to detect malware attacks. This method consists of a deep learning CNN algorithm to detect CNN-DMA malware. The proposed method uses the Malimg dataset. The evaluation results have shown that the proposed method performed well with an accuracy of 99%. Lee *et al.* [5] introduced a feature selection method based on genetic algorithm. In this method, the best features are selected using the genetic algorithm. This method has been evaluated using the Andro-Autopsy dataset. The simulation results have shown that the algorithm has the highest accuracy with an accuracy of 0.981. Yang and *et al.* [6] introduced a hybrid method based on decision tree algorithms and support vector machine. This method has been evaluated using the Drebin dataset. The evaluation results have shown that the proposed method with an accuracy of 0.960 has a higher accuracy than the other methods.

Table 1. Malware attack detection using data mining methods.

Authors	Method	Accuracy	Advantage	Disadvantage
H. Alamro *et al.* [1]	Used LS-SVM, KELM, and RRVFLN methods to identify malware	0.989	By increasing the accuracy, the amount of execution time and computational complexity is reduced.	Failure to protect user privacy
R. Yumlembam *et al.*[2]	Used GNN algorithm to select features and uses GAN algorithm for classify malware.	Drebin: 0.984 CICMaldroid: 0.978	It can help fight malware through retraining	Unbalancing the dataset and finding the best feature in a long time
Kim *et al.* [3]	Used CNN for feature selection, and used lightweight classification to identify malware.	0.989	Identify any known and unknown malware	High execution time due to layered CNN algorithm
A. Anand *et al*.[5]	Used CNN algorithm to detect malware	0.99	Due to the combination of deep learning algorithms, it has higher accuracy.	Training and testing time is high due to multi-layered implementation.
Lee *et al*. [6]	Used genetic algorithm for feature selection	0.981	Reduce execution time.	Failure to increase identification accuracy by feature selection using genetic algorithm with dynamic and static elements.
M. Yang and *et al.* [7]	Used diction tree and SVM algorithms for detect malware	0.960	The combination of two algorithms has been able to increase accuracy and prevent excessive processing.	Failure to compare the proposed method with other recent combined methods.

3. Proposed Method

In this paper, a new method based on data mining algorithms for detecting malware is presented. This method is based on four algorithms: Simulated annealing algorithm (SA), support vector machine (SVM), genetic algorithm (GA), and K-means. Relevant datasets for malware detection have large datasets with a large number of features and records. In this paper, firstly, in order to improve the efficiency of algorithms, to understand the data correctly to gain knowledge about the identification process, to reduce data, to limit the storage process, to reduce costs, to reduce the set of features from the combination of simulated annealing algorithm (SA), support vector machine (SVM) is used for feature selection that has been the improved method in [8] and In the next step is used to identify malware using a combination of algorithms GA and K-means which is almost similar to a method described in [9]. In this section the best features in the initial pipeline are selected, and then the selected features as the initial population of chromosomes are sent to the genetic algorithm in the second pipeline, and here by improving the population of chromosomes and clustering them, malware is identified. In the following, each of these methods has been fully explained. Algorithm 1 shows the pseudocode of the proposed method.

Algorithm 1: SA-SVM-GA-K-Means

```
Input: Initialization of population
Output: Best-features
Select subset feature randomly
While I <= max iteration
    For each search factor
        Calculate the fitness function by logistic regression classification
        LocalSearch: Keep track of overall best metric so far
            Update parameters dataframe // metric, best_metric,
            T = T_0
            Get ascending range indices of all columns
            Create an initial random subset based on 50% of the columns
            Change the current subcategory to create a new subcategory
            Temperature reduction
            Re-evaluation of feature subsets using a regression algorithm
            I = I+1
Return Best features
While I <= max iteration
    For each search factor
        Calculate the fitness function
        Selection of pairs of chromosomes with fitness value to
        Perform crossover operations on selected
        Performing mutation operations Replacing the old generation with the new generation
        Center_Cluster = Best_Choromosome
        Clustering of dataset using K-means algorithm
Calculate the distance of cluster centers with all datasets
    I= I+1
Return Center_Cluster
End
```

3.1. Feature Selection

Most of the datasets present defects in high dimensions, including missing data, string type data, and data in different dimensions. In this paper, due to the use of data mining algorithms, datasets with numerical records are needed, and to obtain better results, missing data are replaced with the average value of each column, and also to place all datasets between two numbers, the Robust Scaling method has been used. This normalization is done using Equation (1)

$$x = \frac{x - Q_2(x)}{Q_3(x) - Q_1(x)} \quad (1)$$

Here features selection is done using the SA-SVM hybrid method. Inside the SVM algorithm, there has been a technique known as the kernel, a non-parametric function that can be used to solve any complex problem without outliers affecting the average of the data. Also, this algorithm avoids overfitting conditions. It does not suffer and works well on generalized data and high-dimensional data, which makes the results have global minimum feature. This algorithm is also used for feature selection due to its low computational complexity and accurate separation of positive and negative points. But determining the number of SVM parameters has a significant impact on classification accuracy, so here SA algorithm has been used to determine SVM parameters for feature selection. Figure 1 shows the process of the proposed method for feature selection.

First, an initial value is randomly generated, in each iteration, point X is considered as the starting point, which is a random vector for choosing the next optimal features of Y. Here, obj(X) is the calculation of the classification accuracy rate of the objective function X, and obj(Y) is the calculation of the classification accuracy rate of the objective function Y, and DE is the difference between obj(X) and obj(Y), which if ΔE < 0, X is replaced by Y, where X is the current solution, and Y is the next solution, given that ΔE < 0, given by $e^{(\frac{□E}{T})}$

The optimal hyperplane has been defined using Eq. (2), where the parameter w has been the weight vector, the parameter x has been the input feature vector, and the parameter b has been the bias. Now it's time to calculate the dividing line, in this case, two parameters w and b have been calculated using Equation (3).

$$W x^{t} + b = 0 \tag{2}$$

$$\max_{W\in H, b\in R} \min_{1\le i\le N} \{\|x - x_i\| \\ |x \in H(w * x) + b = 0|, i = 1 \ldots m\} \tag{3}$$

The separating hyperplane has an optimal separating hyperplane (OSH) that contains the largest distance between two points on its two sides. Equation (4) has been used to calculate the distance between two points of the support vector.

$$\frac{1}{\|w\|^2} \tag{4}$$

Using this equation, the Lagrange polynomial can be minimized. α is defined as sequence $(\alpha_1 . \alpha_2 \ldots \alpha_m)$. Now Lagrange's polynomial has been combined with Equation (4) and the maximization Equation (5) has been obtained.

$$w(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} \alpha_i \alpha_j y_i y_j \\ (x_i, x_j) \tag{5}$$

If there has been an expansion in the limit of Equation (5), the function of hyperpage has been calculated using Equation (6).

$$f(x) = \operatorname{sgn}(\sum_{i=1}^{m} y_i \alpha_i \langle x, x_i \rangle + b) = 0 \tag{6}$$

When the data cannot be linearly separated, the data is mapped to a higher dimensional feature space. OSH is a built-in feature space. In this situation, the feature space vectors are evaluated based on the kernel k, the kernel function is applied to the input data, and the weight vector is converted into an extension in the feature space and calculated using Equation (7).

$$f(x) = \operatorname{sgn}(\sum_{i=1}^{m} y_i \alpha_i k \langle x_i, x_j \rangle + b) \tag{7}$$

Now the kernel function has helped the SVM algorithm to find the optimal solution. SA has an optimization algorithm based on solids annealing in metallurgy, which is suitable for high-dimensional problems. This method involves gradually cooling a given material from a high temperature in a controlled manner to reduce its defects. There is a lot of similarity between minimizing a cost function and slowly cooling a material to a ground state that has little energy. How to change a thermodynamic system from the x_{old} state to the x_{new} state is done using Equation (8).

$$P = \begin{cases} 1 \rightarrow if E(X_{new}) < E(X_{old}) \\ \exp \frac{E(X_{new}) - E(X_{old})}{T} \rightarrow if E(X_n ew) \ge E(X_o ld)) \end{cases} \tag{8}$$

where parameter T has been temperature, and E(x_{new}) and E(x_{old}) have been system energy in X_{new} and X_{old} states. In the case of creating inverse model-based damage detection that minimizes the parameters for damage detection and finds the differences between the measured and calculated modal characteristics where the parameter f (B) is an objective function, the parameter N_e has been the number of elements, and the parameter B contains the stiffness reduction coefficients. It is assumed that the elemental mass matrix does not change, and the stiffness coefficient has been the damaged structure. The stiffness matrix has been calculated as the sum of the damaged stiffness matrices using Equation (9).

$$K = \sum_{n=1}^{N_e} (1 - \alpha_n)^2 \tag{9}$$

in which k_n parameter is a hardness matrix of the n element and the value of the α_n parameter is in the range of 0 to 1 and it shows the severity of the damage. One of the objective functions for damage detection has been calculated using Equation (10).

$$f(B)=\sqrt{\frac{1}{m}{}^{m}_{(i=1)}\sum_{i=1}^{m} w\frac{w_i^{measured}-w_i^{calculated}}{w_i^{measured}}^{2}} \quad (10)$$

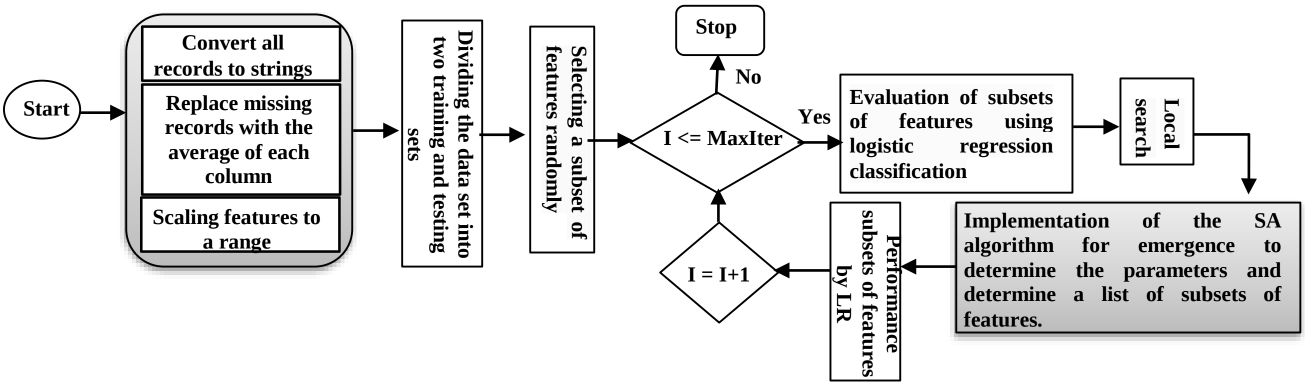
Start
Convert all records to strings
Replace missing records with the average of each column
Scaling features to a range
Dividing the data set into two training and testing sets
Selecting a subset of features randomly
Stop
No
I <= MaxIter
Yes
Evaluation of subsets of features using logistic regression classification
Local search
Implementation of the SA algorithm for emergence to determine the parameters and determine a list of subsets of features.
Performance subsets of features by LR
I = I+1

Figure 1. The proposed SA-SVM feature selection hybrid method

3.2. Attack Detection

In this section, the combined method based on genetic method (GA) and K-means clustering has been used to identify malware in this method, the features selected in the previous step were selected as primary chromosomes and clustering was done, and then the malware was identified using the K-means algorithm. The genetic algorithm can well support the multi-objective process in a Here, the fitness of each chromosome in the current population has been calculated using the fitness function based on the k-NN algorithm. The k-NN algorithm has calculated the fit using the Euclidean distance of the shortest distance between the test and the training set in the feature space. The Euclidean distance method is calculated using Equation (11).

$$D(X_{test}X_{train})=\sqrt{\sum_{m=1}^{M}(X_{test}-X_{l})} \quad (11)$$

Chromosome with attribute value "1" is selected and chromosome with attribute value "0" is not selected to evaluate the corresponding chromosome. At each step of the iteration, the current population is calculated using Equation (12).

$$fit=\frac{\alpha}{N_f}+\exp(\frac{-1}{N_f}) \quad (12)$$

The GA selection operator in natural selection increases the chance of survival, which causes the reproduction of their genes to the next generation. Here the roulette wheel is used for the selection process. In this method, the lower and upper limits of the roulette wheel are 0 and 1, respectively, and people who have a higher fitness value have a higher chance of being selected by the roulette

probabilistic and random manner, which is suitable for discrete and continuous data. But it has a lot of calculations on high-dimensional datasets, which are used here to narrow the search boundaries of the K-means algorithm. In the following, each of the steps has been fully explained. Figure 2 shows the process of the proposed method for malware detection.

wheel. The advantage of this method compared to other methods is that people with the least physical fitness can mate and enter the new generation. The probability of the roulette cycle is done using the Equation (13).

$$P_i=\frac{f_i}{\sum_{j=1}^{N} f_i} \quad (13)$$

Selection in the previous stage has selected the parents for the crossover stage while in this stage the genes are exchanged between individuals to produce new solutions. Here, the chromosomes are divided into two parts, then the genes are exchanged between the two chromosomes. If the solutions get stuck in the local optimal solutions, the crossing of new chromosomes with new genes different from the parents' genes has not resulted. In this situation, the mutation operator is used, which causes random changes in genes. Using the ability parameter, the probability of mutation (Pm) for each gene in the child chromosome in the crossover stage has a number in the range [0,1]. In this operation, minor changes have occurred in some randomly selected genes. After selecting the optimal chromosomes, in the next step, the K-means algorithm was entered, and the optimal chromosomes were selected as clustered centers and randomly clustered. Now, in order to minimize the degree of similarity within the clusters, the distance of each particle with the centers of each cluster. By maximizing the similarity between the clusters, the clustering has been completed, now the malware has been identified using the K-means algorithm. Also, the complexity analysis of SA-SVM-GA-K-means is presented for the readers. The SA and SVM algorithm have O($\sqrt{n}$) and O(n^3) complexity, respectively. In the proposed method, despite the implementation of two algorithms, but due to the setting of the SVM parameters to SA by selecting the features using the SVM algorithm and optimizing its parameters using the SA algorithm, its computational complexity is reduced to O(n^2). The GA and K-means algorithm have O(n^3) and O(n) complexity, respectively. In this method, the GA algorithm, which is a method with a high execution time, is used due to the reduction of the search boundaries using the Cummins algorithm as a result of the time complexity of the whole proposed method is O(n^4).

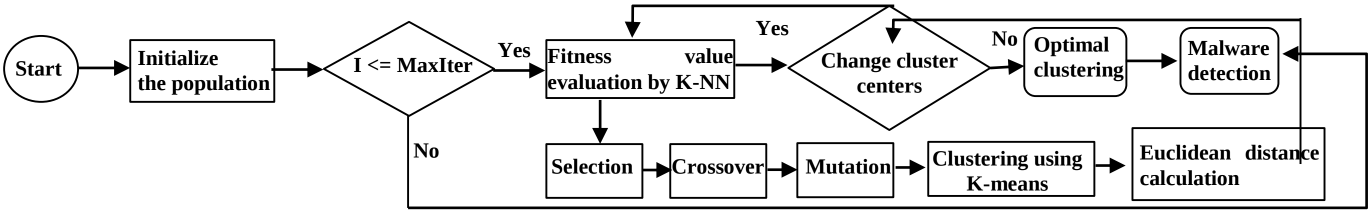

Figure 2. The proposed GA-K-means malware detection hybrid method.

4. Experimental Results

Here, a detection method is proposed to identify any malware that still exists at an optimal time and with high accuracy. Which are used here to help identify malware with high accuracy and low error in a short time using the proposed SA-SVM feature selection method. Also, the GA-K-Means hybrid method has proposed to identify malware in a dataset. To evaluate the performance of the proposed method, experiments are performed in a Python program and on two datasets Andro-Autopsy and CICMalDroid 2020. Also, to show the superiority of the proposed method in detecting malware compared to other methods, they have compared with 7 other methods such as PSO-C4.5, SVM K_Modes, K-means, Learning Vector Quantization (LVQ), XGBoost, and GA. Feature selection in all three methods is done at 100 points and after that, no improvement was made in the feature selection process. The highest accuracy of feature selection using the SA method is (0.965 and 0.970), using the SVM method (0.970 and 0.969), and the proposed method (0.984 and 0.988). This indicates that the proposed method for feature selection in both datasets is able to select the best features that have important information, independent of each other, and remove noisy features, reduce the dimensions of the dataset, storage costs, and the amount of computation, and increase the processing speed.

Table 2 shows the accuracy of identifying attacks using data mining methods and the proposed method (GA-K-means) in both datasets using features selected by SA, SVM, and the proposed method (SA-SVM). Attack detection accuracy is calculated using Equation (14). According to this Table, the attack detection accuracy of all methods using the proposed method for feature selection is able to help the algorithms understand the data to identify suitable patterns by selecting the main features and has also improved the efficiency of the algorithms. According to the same Table, the proposed method (GA-K-means) can detect attacks in all features selected by all three methods and in both datasets with accuracies (0.942, 0.973, and 0.995) in the Andro-Autopsy dataset. With accuracies (0.923, 0.970, and 0.989) in the CICMalDroid 2020 dataset, it has been able to identify malware with the lowest error rate and the highest accuracy rate in all three methods.

$$Accuracy = TP + \frac{TN}{TP} + FP + TN + FN \tag{14}$$

In Table 3, the proposed method which is based on criteria such as negative predictive value (NPV), false positive rate (FPR), false negative rate (FNR), true negative rate (TNR), true positive rate (TPR), and positive predictive values, (PPV) has been evaluated. These criteria are the basis of other evaluation criteria and are calculated using the following equations.

$$NPR = \frac{TN}{TN + FN} \tag{15}$$

$$FPR = \frac{FP}{FP + TN} \tag{16}$$

$$FNR = \frac{FN}{FN + TN} \tag{17}$$

$$TPR = \frac{TP}{TP + FP} \tag{18}$$

Table 2. Classification accuracy of compared algorithms.

Dataset	Algorithms	SA	SVM	SA-SVM
Andro-Autopsy	PSO-C4.5	0.890	0.90	0.950
	SVM	0.90	0.85	0.903
	K-Modes	0.905	0.904	0.991
	K-Means	0.932	0.899	0.981
	LVQ	0.765	0.750	0.9
	XGBoost	0.895	0.863	0.955
	GA	0.90	0.935	0.961
	GA-K-means	0.942	0.973	0.995
CICMalDroid 2020	PSO-C4.5	0.699	0.801	0.861
	SVM	0.890	0.863	0.917
	K-Modes	0.90	0.932	0.987
	K-Means	0.565	0.798	0.815
	LVQ	055	0.632	0.691
	XGBoost	0.432	0.450	0.50
	GA	0.800	0.899	0.973
	GA-K-means	0.923	0.970	0.989

$$FNR = \frac{TN}{TN + FP} \tag{19}$$

$$PPV = \frac{TP}{TP + FP} \tag{20}$$

In Table 3, the proposed method (GA-K-means) in both datasets with the highest TPR (0.964 and 0.985) and TNR (0.985 and 0.989) and the lowest FPR (0.043 and 0.022) and FNR (0.036 and 0.015) has been able to detect malware.

Table 4 summarizes the results of how to identify the proposed method. In this Table, the criteria of precision, F-measure, recall, specificity, and sensitivity are used. These criteria are calculated using the following equations.

$$\Pr ecistion = \frac{TP}{TP + FP} \tag{21}$$

$$\mathrm{Re}\,call = \frac{TP}{TP + FN} \tag{22}$$

$$F1 - measure = 2 * \frac{\Pr ecision * \mathrm{Re}\,call}{\Pr ecision + \mathrm{Re}\,call} \tag{23}$$

$$Specificity = \frac{TN}{TN + FP} \tag{24}$$

$$Sensivity = 1 - FNR \tag{25}$$

The F-measure is a combination of precision and recall. Precision is the percentage of samples that are positive and correctly classified as positive. Remembering the percentage of positive cases that are correctly predicted as positive. This measure has symmetrically demonstrated both precision and recall. The highest possible value for this criterion is 1, indicating that both precision and recall were perfect, and the lowest possible value is 0, indicating zero precision or recall. In this table, the proposed method with F-measure values (0.989 and 0.978) has been able to identify positive cases with the least amount of error.

Table 3. Comparing value confusion matrix proposed method.

Dataset	Algorithm	NPV	FPR	FNR	TPR	TNR	PPV
Andro-Autopsy	PSO-C4.5	0.873	0.050	0.110	0.89	0.949	0.051
	SVM	0.222	0.027	0.129	0.871	0.973	0.375
	K-Modes	0.204	0.022	0.120	0.88	0.977	0.206
	K-Means	0.524	0.025	0.099	0.901	0.974	0.122
	LVQ	0.236	0.125	0.081	0.919	0.875	0.333
	XGBoost	0.424	0.088	0.133	0.867	0.933	0.173
	GA	0.380	0.125	0.110	0.89	0.90	0.145
	GA-K-means	0.245	0.022	0.036	0.964	0.985	0.325
CIC Mal Droid 2020	PSO-C4.5	0.204	0.218	0.08	0.92	0.781	0.407
	SVM	0.320	0.0435	0.118	0.882	0.956	0.254
	K-Modes	0.519	0.040	0.095	0.905	0.959	0.125
	K-Means	0.569	0.161	0.285	0.715	0.838	0.106
	LVQ	0.365	0.261	0.4	0.6	0.838	0.215
	XGBoost	0.195	0.299	0.366	0.634	0.421	0.423
	GA	0.145	0.202	0.256	0.744	0.798	0.125
	GA-K-means	0.256	0.043	0.015	0.985	0.989	0.343

In Table 5, the proposed method is compared with other new and advanced hybrid methods based on meta-heuristic and machine learning algorithms in both Andro-Autopsy and CICMalDroid 2020 datasets, and in Table 6, the proposed method is compared with other new hybrid methods and advanced algorithms based on meta-heuristics and machine learning are compared without data bias. According to both tables, the proposed method is more efficient than other proposed methods in recent works due to its high accuracy, true positives and negatives, and low false negatives, false positives, and errors.

Table 4. summarizes the results of identify proposed method.

Dataset	Algorithm	Precision	F-measure	Recall	specificity	sensitivity
Andro-Autopsy	PSO-C4.5	0.285	0.444	0.990	0.980	0.975
	SVM	0.985	0.924	0.870	0.972	0.870
	K-Modes	0.985	0.992	0.990	0.977	0.990
	K-Means	0.937	0.967	0.991	0.974	0.991
	LVQ	0.900	0.900	0.9	0.875	0.918
	XGBoost	0.882	0.937	0.985	0.978	0.990
	GA	0.923	0.923	0.989	0.980	0.991
	GA-K-means	0.991	0.989	0.999	0.99	0.999
CICMalDroid 2020	PSO-C4.5	0.851	0.884	0.920	0.781	0.920
	SVM	0.956	0.917	0.882	0.956	0.881
	K-Modes	0.904	0.90	0.904	0.959	0.904
	K-Means	0.5	0.588	0.714	0.838	0.714
	LVQ	0.735	0.688	0.691	0.838	0.6
	XGBoost	0.214	0.352	0.988	0.992	0.990
	GA	0.943	0.875	0.82	0.908	0.989
	GA-K-means	0.973	0.978	0.988	0.991	0.999

Table 5. Performance comparison with other works in two datasets Andro-Autopsy and CICMalDroid 2020 (%).

Authors	Method	Dataset	Accuracy	Precision	Recall	F1-score
H. ALAMRO et al. [1]	AAMD + OELAC	Andro-Autopsy	0.989	0.991	0.989	0.990
K. Sharma et al. [10]	RNPDroid	Andro-Autopsy	0.974	0.979	0.971	0.981
Lee *et al. [*6]	Ga + MLP	Andro-AutoPsy	0.984	-	-	0.976
Proposed Method	***SA-SVM-GA-K-means***	***Andro-Autopsy***	***0.995***	***0.991***	***0.999***	***0.989***
D. Aboshady[11]	APKOWL	CICMalDroid 2020	0.970	0.975	0.990	0.980
R. Manzil et al. [12]	RF + Hufman encoding	CICMalDroid 2020	0.931	0.931	0.931	0.931
C. Avci.et al. [13]	CNN + LSTM	CICMalDroid 2020	0.8857	0.454	0.086	0.142
V. Lavanya et al. [14]	WDCNN + EROA	CICMalDroid 2020	0986	0.961	0.961	0.961
Proposed Method	***SA-SVM-GA-K-means***	***CICMalDroid 2020***	***0.989***	***0.973***	***0.988***	***0.978***

Table 6. Performance comparison with other works (%).

Authors	Method	Dataset	Accuracy	Precision	Recall	F1-score
L. Potharlanka et al. [15]	PSO + FA, + WOA	PROMISE NASA	0.913 0.965	0.93 0.975	0.958 0.958	- -
K. Keserwani et al. [16]	GA + DNN	UNSW-NB15	0.981	0.981	0.981	0.981
F. Taher et al. [17]	Fuzzy + HHO + ANN	Drebin CICAndMal2017 APKMirror VirusShare	0.973	0.967	0.967	0.971
R. Yumlembam *et al.*[2]	GNN + GAN	Drebin CICMaldroid	0.984 0.978	0.929 0.987	0.910 0.984	0.919 0.986
Kim *et al.* [3]	CNN +light weight	Google Play Store	0.912	-	-	-
A. Anand *et al.*[5]	CNN+ DMA	Malimg	0.990	0.958	0.989	-
M. Yang and *et al.* [7]	DT-SVM	-	0.960	0.960	0.960	0.960
Proposed Method	***SA-SVM-GA-K-means***	***Andro-Autopsy***	***0.995***	***0.991***	***0.999***	***0.989***
Proposed Method	***SA-SVM-GA-K-means***	***Andro-Autopsy***	***0.995***	***0.991***	***0.999***	***0.989***

The time complexity of an algorithm is the amount of time an algorithm takes to run, which has a function of the length of the input. Table 7 shows the computational complexity of the proposed method and the compared methods on the Andro-Autopsy dataset. The execution time of all methods are evaluated using 15 features, 1000 data and one execution round. According to this Table, the ascending order of computational complexity of algorithms includes static, dynamic, functional and interactive algorithms. The proposed algorithm consists of four algorithms, the computational complexity of each of which has polynomial, in this method, both SA and SVM algorithms are implemented in the first phase. Although the SA algorithm helped in determining the SVM parameters, it is a significant effect in reducing the complexity. In the second stage, GA and K-means algorithms have been used for malware identification. Due to the reduction of the search boundaries and the reduction of the number of cycles, using the K-means algorithm has a great impact on reducing the time complexity of the algorithm. In this way, it has tried to reduce the amount of computational complexity and increase its accuracy compared to other methods.

Table 7. Computational complexity compared methods.

Algorithms	Time complexity	n=15
PSO-C4.5	O(n^2 log 2n)	1125
SVM	O(n^3)	3375
K-Modes	O(n)	15
K-Means	O(n)	15
LVQ	O(Pn)	15000
XGBoost	O(kd$\|x\|logn$)	1654
GA	O(n^3)	3375
SVM-SA-GA-K-means	O(n^4)	256

5. Conclusion

In this paper, a hybrid detection method based on data mining methods for detecting malware is presented. This method is a combination of four algorithms: simulated annealing algorithm (SA), support vector machine (SVM), genetic algorithm (GA) and K-means. First, suitable features have been selected; using SVM algorithm and SVM algorithm parameters have optimized using SA algorithm and prevent the proposed method from getting stuck in the local optimum. Then, using GA-K-means method, malwares were identified. In this method, a GA algorithm with crossover and mutation operations causes diversity in the population and is also able to produce a list of optimal solutions due to competition, which is improved over time. In this method, the chromosomes produced by the GA algorithm are selected as the cluster centers of the K-means algorithm. This has reduced the search boundaries and is suitable for high-dimensional datasets, and of course, this algorithm saves a large number of chromosomes. It has a high storage cost, and using the K-means algorithm and determining the number of chromosomes can reduce the storage cost. The proposed method has been evaluated using two datasets of Andro-Autopsy and CICMalDroid 2020. Moreover, to show the improvement of the proposed method, it was compared with 6 other methods. The evaluation results showed that the proposed method improved with accuracy (0.995 and 0.989) and the lowest mean squared error (0.014 and 0.015) in both datasets compared to other methods.

Reference

[1] H. Alamro, W. Mtouaa, S. Aljameel, A.S. Salama, M.A. Hamza, and A.Y. Othman, "Automated android malware detection using optimal ensemble learning approach for cybersecurity," IEEE Access, 2023.

[2] R. Yumlembam, B. Issac, S.M. Jacob, and L. Yang, "Iot-based android malware detection using graph neural network with adversarial defense," *IEEE Internet of Things Journal*, 2022.

[3] J. Kim, Y. Ban, E. Ko, H. Cho, and J.H. Yi, "MAPAS: a practical deep learning-based android malware detection system," *International Journal of Information Security*, vol. 21, no. 4, pp. 725-738, 2022.

[4] R. Morshedi, S.M. Matinkhah, and M.T. Sadeghi. "Intrusion Detection for IoT Network Security with Deep learning." *Journal of AI and Data Mining* (2024).

[5] A. Anand, S. Rani, D. Anand, H. M. Aljahdali, and D. Kerr, "An efficient CNN-based deep learning model to detect malware attacks (CNN-DMA) in 5G-IoT healthcare applications," *Sensors*, vol. 21, no. 19, pp. 6346, 2021.

[6] J. Lee, H. Jang, S. Ha, and Y. Yoon, "Android malware detection using machine learning with feature selection based on the genetic algorithm," *Mathematics*, vol. 9, no. 21, pp. 2813, 2021.

[7] M. Yang, X. Chen, Y. Luo, and H. Zhang, "An android malware detection model based on dt-svm," *Security and Communication Networks*, vol. 2020, pp. 1-11, 2020.

[8] S.W. Lin, Z.J. Lee, S.C. Chen, and T.Y. Tseng, "Parameter determination of support vector machine and feature selection using simulated annealing approach," *Applied soft computing*, vol. 8, no. 4, pp. 1505-1512, 2008.

[9] K. Krishna and M.N. Murty, "Genetic K-means algorithm," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 29, no. 3, pp. 433-439, 1999.

[10] K. Sharma and B.B. Gupta, "Mitigation and risk factor analysis of android applications," *Computers & Electrical Engineering*, vol. 71, pp. 416-430, 2018.

[11] D. Aboshady, N.E. Ghannam, E.K. Elsayed, and L. Diab, "APKOWL: An Automatic Approach to Enhance the Malware Detection," *Mobile Networks and Applications*, pp. 1-12, 2023.

[12] H.H.R. Manzil and S. Manohar Naik, "Android malware category detection using a novel feature vector-based machine learning model," *Cybersecurity*, vol. 6, no. 1, p. 6, 2023.

[13] C. Avci, B. Tekinerdogan, and C. Catal, "Analyzing the performance of long short-term memory architectures for malware detection models," *Concurrency and Computation: Practice and Experience*, vol. 35, no. 6, pp. 1-1, 2023.

[14] V.Lavanya and P.C. Sekhar, "Efficient Cybersecurity Model Using Wavelet Deep CNN and Enhanced Rain Optimization Algorithm," *International Journal of Image and Graphics*, p. 2450048, 2023.

[15] J.L. Potharlanka, "Feature importance feedback with Deep Q process in ensemble-based metaheuristic feature selection algorithms," *Scientific Reports*, vol. 14, no. 1, p. 2923, 2024.

[16] P. K. Keserwani, M.C. Govil, and E.S. Pilli, "An effective NIDS framework based on a comprehensive

survey of feature optimization and classification techniques," *Neural Computing and Applications*, vol. 35, no. 7, pp. 4993-5013, 2023.

[17] F. Taher, O. AlFandi, M. Al-kfairy, H. Al Hamadi, and S. Alrabaee, "DroidDetectMW: A Hybrid Intelligent Model for Android Malware Detection," *Applied Sciences*, vol. 13, no. 13, p. 7720, 2023.

رویکرد یادگیری ماشین ترکیبی و الگوریتم ژنتیک برای تشخیص بدافزار

مهدیه معاذاللهی و سوده حسینی*

گروه علوم کامپیوتر، دانشکده ریاضی و کامپیوتر، دانشگاه شهید باهنر کرمان، کرمان، ایران.

ارسال ۲۰۲۳/۱۱/۳۰؛ بازنگری ۲۰۲۴/۰۱/۲۴؛ پذیرش ۲۰۲۴/۰۳/۰۹

چکیده:

شناسایی و جلوگیری از آلودگی‌های بدافزار در سیستم‌ها به یک ضرورت حیاتی تبدیل شده است. در این مقاله یک روش ترکیبی برای تشخیص بدافزار، با استفاده از الگوریتم‌های داده کاوی مانند بازپخت شبیه‌سازی شده (SA) ، ماشین بردار پشتیبانی(SVM) ، الگوریتم ژنتیک(GA) ، و K-means ارائه شده است. روش پیشنهادی این الگوریتم‌ها را برای دستیابی به تشخیص بدافزار موثر ترکیب کرده است، در ابتدا، روشSA-SVM برای انتخاب ویژگی استفاده شده، که در آن الگوریتم SVM بهترین ویژگی‌ها را شناسایی کرده است و الگوریتم SA پارامترهای SVM را محاسبه کرده است. دوم از روش GA-K-means برای شناسایی حملات استفاده شده است که الگوریتم GA بهترین کروموزوم را برای مراکز خوشه ای انتخاب کرده و الگوریتم-K means برای شناسایی بدافزارها اعمال شده است. برای ارزیابی عملکرد روش پیشنهادی، از دو مجموعه دادهAndro-Autopsy و CICMalDroid 2020 استفاده شده است. نتایج ارزیابی نشان داده است که روش پیشنهادی به نرخ‌های مثبت واقعی بالا (۰٫۹۶۴، ۰٫۹۸۵)، نرخ‌های منفی واقعی (۰٫۹۸۵، ۰٫۹۸۹)، نرخ‌های منفی کاذب پایین (۰٫۰۳۶، ۰٫۰۱۵) و نرخ‌های مثبت کاذب (۰٫۰۲۲، ۰٫۰۴۳) دست یافته است. نتایج ارزیابی نشان داده است که این روش به طور موثری بدافزارها را شناسایی کرده است و در عین حال به طور منطقی شناسایی‌های نادرست را به حداقل رسانده است.

کلمات کلیدی: تشخیص بدافزار، روش ترکیبی، الگوریتم‌های داده‌کاوی، انتخاب ویژگی.