Journal of Ubiquitous Computing and Communication Technologies (UCCT) (2021)
Vol.03/ No.02
Pages: 135-149
https://www.irojournals.com/jucct/
DOI: https://doi.org/10.36548/jucct.2021.2.006

Design an Adaptive Hybrid Approach for Genetic Algorithm to Detect Effective Malware Detection in Android Division

B Vivekanandam
Senior Lecturer, Faculty of Computer Science and Multimedia, Lincoln University College, Malaysia
E-mail: vivekanandam@lincoln.edu.my

Abstract

Data pre-processing is critical for handling classification issues in the field of machine learning and model identification. The processing of big data sets increases the computer processing time and space complexity while decreasing classification model precision. As a result, it is necessary to develop an appropriate method for selecting attributes. This article describes a machine learning technique to solve functional selection by safeguarding the selection and mutation operators of genetic algorithms. During population calculations in the training set, the proposed method is adaptable. Furthermore, for various population sizes, the proposed method gives the best possible probability of resolving function selection difficulties during training process. Furthermore, the proposed work is combined with a better classifier in order to detect the different malware categories. The proposed approach is compared and validated with current techniques by using different datasets. In addition to the test results, this research work utilizes the algorithm for solving a real challenge in Android categorization, and the results show that, the proposed approach is superior. Besides, the proposed algorithm provides a better mean and standard deviation value in the optimization process for leveraging model effectiveness at different datasets.

Keywords: Genetic algorithm, android malware detection process

ISSN: 2582-337X
Submitted: 26.05.2021
Revised: 29.06.2021
Accepted: 19.07.2021
Published: 23.07.2021

1. Introduction

Over the last decade, the proliferation of smartphones has resulted in an unprecedented increase in their ability to store ownership data, which has become much more sensitive and private than simple contact information and social media information, including passwords for usernames, financial site and payment information, including credit card and bank numbers [1, 2]. The growing use of smartphones and increasing popularity of the Android operating system have brought attention to those who can achieve their destructive goals by developing a malware. It is extremely simple to access your smartphone since thousands of Android apps are available on variety of marketplaces [3]. Figure 1 shows the simplified block diagram of malware detection.

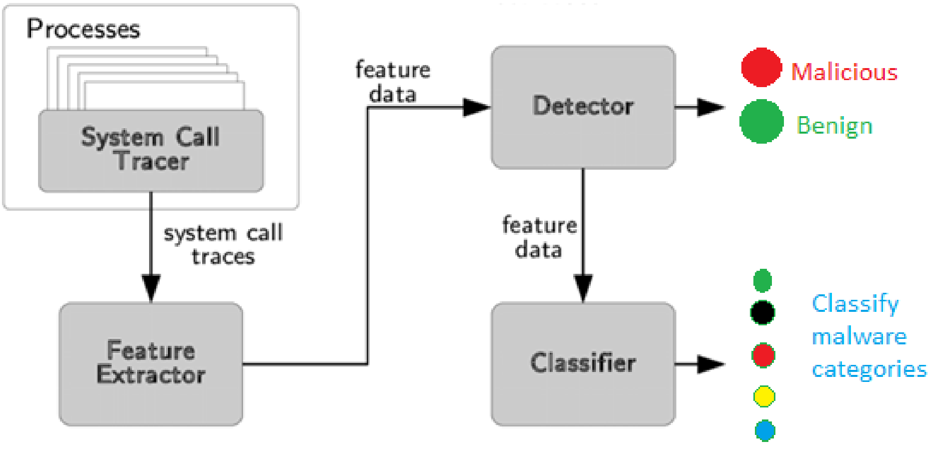

Figure 1. Simplified Block Diagram of Malware Detection

It is because anybody may distribute an application to the public via several markets without demonstrating whether the software is benign or harmful to the operating system of the end users. [4] demonstrates that Android is used in a wide range of companies and demographic groups; according to current evidence, 73% of smartphones run Android [5], indicating that the

ISSN: 2582-337X
Submitted: 26.05.2021
Revised: 29.06.2021
Accepted: 19.07.2021
Published: 23.07.2021

OS is used by a wide range of businesses and segments of the public. As a result, an efficient solution is required to prevent malware from spreading in Android devices [6]. Identifying malware may include acquiring and converting data such as permissions, API requests, and network addresses into vehicle space, among other things. There are three sorts of characteristics and that may be classified as follows:

1. Filtering
2. Wrapping
3. Incorporation

Before treatment, filtering methods are employed for sorting functions, with functions chosen with higher rankings depending on their ranking positions [7]. Criteria for selecting characteristics in wrapper approaches are used to create wrapper methods to forecast search algorithm performance assist predicting one of the search algorithms to identify the suitable subset. Integrated techniques may be utilized without separating the data into training and test sets by choosing a variable during the workout [8].

With the increasing security hazard posed by Android devices, identifying the malicious Android-based malware in real-time is becoming increasingly essential. To maintain the malware's constant adoption of new technologies and methods, Android malware detection utilizes a variety of algorithms and technologies on a continuous basis, in accordance with the malware's ongoing adoption of new technology and techniques [9, 10]. Initial efforts have been made to develop the signature-based detection method (which is the same as the personal computer system). There are no techniques for identifying unknown dangerous software, and an increasing number of individuals are researching into dynamic behavior-based methods of characteristic identification [11, 12].

2. Organization of the Research

The rest of the research article is organized as follows: Section 3 presents current research on the technique used for detecting the Android malware. Section 4 deals with the planned work of the proposed approach. Section 5 describes the outcomes gained from the proposed work. Section 6 concludes our study with more improvement.

3. Preliminaries

In recent years, researchers have attempted to include engineering and data mining approaches in the detection of Android malware, mostly through producing massive malware data sets. Take the example of Yang Huan et al work [13] has created a multi-characteristic system of detectors by using a three-layer, hybrid ensemble approach, and Amos et al [14] proposed a STREAM framework to quickly verify the Android malware classification.

Zhang et al used different forms of Bayesian algorithms, have selected the features through permissions, code properties, and a combination of the two with the Information Gain algorithm (IG). The highest accuracy achieved is 93%, which is obtained by combining the two features [15]. The authors Yerima et al used different learning techniques for the features obtained through the static method; they have used three algorithms, such as IG, Fisher Score, and Chi-square for feature selection [16]. Shabtai et al provided a permission-based technique that employed various machine learning algorithms, the best of which yielded an 86% using the Random Forest algorithm [17]. Sanz borja et al have utilized a variety of machine learning algorithms, and the highest accuracy obtained with the SVM algorithm is 97%, which has been identified through malware API calls and permission features [18, 19]. In recent research, authors Peiravian Naser et al acquired a 98% by employing a genetic algorithm to search for permission-based characteristics, and they identify malware using machine learning techniques. The author suggested a method that is quite similar

ISSN: 2582-337X
Submitted: 26.05.2021
Revised: 29.06.2021
Accepted: 19.07.2021
Published: 23.07.2021

to genetic algorithm searches for feature extraction in order to improve model parameter accuracy [20].

With this difference, in addition to the permissions, they also use application components to detect Android malware. Also study of yildiz oktay et al offers a simple and effective way to select features, based on the number of times each API call is used. The features are sorted into two lists, the first list includes the names and number of API calls in malware and the second list includes the names and number of API calls in benign applications. The duplicate API calls of the two lists are removed in one option, while comparable API calls remain intact with the other. The characteristics are therefore prepared for classification algorithms [21, 22]. The asexual permutation genetic algorithm is based on the discovery of Barbara McClintock of the DNA sequence structure in the 1950s was developed by Anabela Simes et al [23]. In contrast to the basic permutation of a sexual mechanism, the genetic algorithm for asexual permutation has only one parent, while a single person includes transposons and inserts locations.

Alireza Farasat et al developed mathematical models to handle optimization and decision-making issues based on the evolution of the process of asexual reproduction. The experimental findings showed a convergence of the method. They found that the asexual reproduction method offers substantial benefits when issue resolution is achieved in real-time by traversing the search area without reducing the convergence length and works best in the PSO's swarm intelligence algorithm [24]. Still the following question can be unanswered in malware detection processing.

1. How do feature rankings vary when allowances, attempts, and API requests are combined?
2. How do feature rankings vary in functional selection algorithms?

ISSN: 2582-337X
Submitted: 26.05.2021
Revised: 29.06.2021
Accepted: 19.07.2021
Published: 23.07.2021

3. How does machine learning model precision vary across techniques and selection features?
4. How does the correctness of the feature size effect model across feature selection techniques?
5. What are the key elements of permissions, attempts, and API calls?

4. Proposed Methodology

Feature selection is an important stage in classification processing and data mining pre-processing with the goal of increasing the classifier's accuracy. To test the suggested method's performance advantages, we utilize it to identify malicious Android applications and provide a framework mechanism for resolving the issue. Figure 2 depicts the proposed overall framework structure.

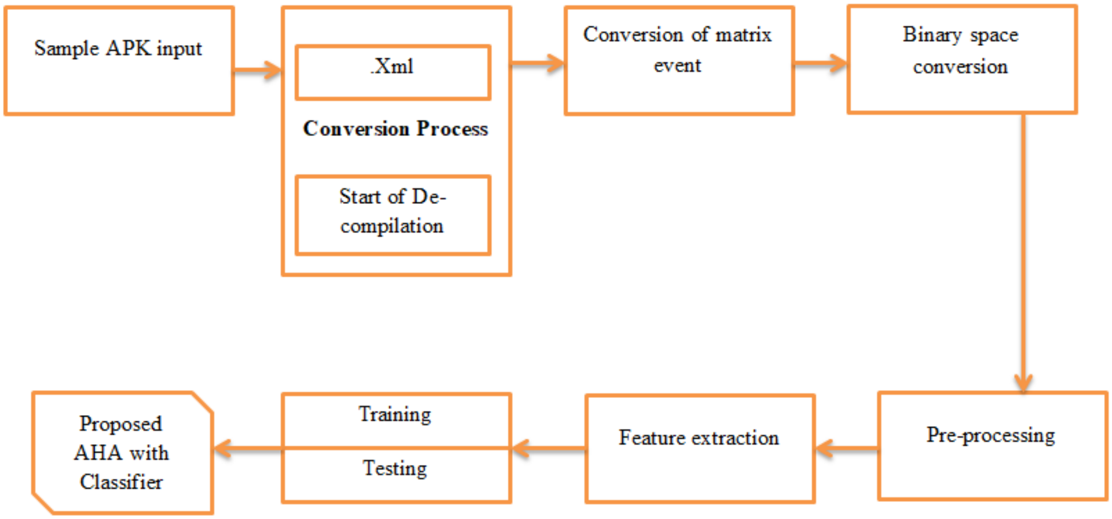

Figure 2. Overall Proposed Framework Structure

ISSN: 2582-337X
Submitted: 26.05.2021
Revised: 29.06.2021
Accepted: 19.07.2021
Published: 23.07.2021

The APK is a summary of an android application package term that is not like source code type division but similar to the file zip format. The decompiler software must be used for pre-processing the APK decompression to access the data present in the package. The Apktool tool used in this research is a lightweight decompile tool, which can decode resources and applications in the fundamental state of java source code in order to analyse the file structure automatically [25]. It may be used locally and supports many platform analysis applications. This apk tool may be used to obtain the file resource list by decompiling a legitimate Android application.

Algorithm:

Step 1:

Make ready the population size with “*N*”, “i” is the actual number of iterations and “f*”* is the total number of features.

Step 2:

Calculate the fitness value of people in the present population and obtain the *F(n)* fitness value for each chromosome by following n computations to obtain the fitness value of each person with various features.

Step 3:

Choice of the selection technique in the AHA for genetic algorithm is tournament selection and the number of people chosen each time is 3; replacement sampling is the selection method. Three people are randomly chosen to calculate their fitness from the population and the better one will join the next generation.

Step 4:

Change the mutation operation is conducted on each chromosome in the population according to a specified mutation probability, and the mutated person is different from those following the selection operation and has distinct features, and these people with different features form a new population.

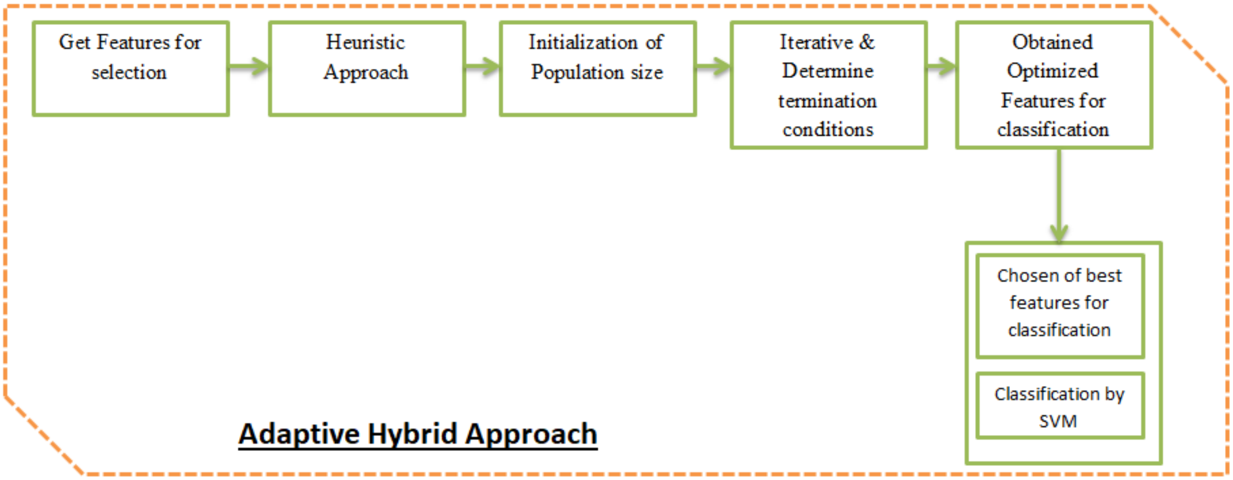
Get Features for selection
Heuristic Approach
Initialization of Population size
Iterative & Determine termination conditions
Obtained Optimized Features for classification
Chosen of best features for classification
Classification by SVM
Adaptive Hybrid Approach

Figure 3. Proposed Framework

Step 5:

The conditions for termination of algorithms are qualitative. If the algorithm ends, the number of iterations will then exit the feature subset chosen by the algorithm and proceed to Step 6 at the conclusion of the iteration. If not, Step2 will be executed.

Step 6:

Given results are provides as output by SVM. The person with the greatest fitness value is the output and the gene position is 1.

ISSN: 2582-337X
Submitted: 26.05.2021
Revised: 29.06.2021
Accepted: 19.07.2021
Published: 23.07.2021

Figure 3 shows block diagram of our specific proposed work. Our proposed module optimises the original data set and selects the function subset that influences the classification effect substantially. The initial iterative group is chosen randomly and then iterated to screen the group through the improved genetic algorithm and finally obtain a number of allow ability functions to optimise the effect of the classification [26]. A feature data set gathered by the decompile module is selected using an asexual genetic algorithm to create a functional part of a classification machine, and its classification accuracy evaluates the Android software classification feature selection process.

5. Results & Discussion

This research article uses AHA's architecture to assess benchmark binary tests to verify the effectiveness and effectiveness of algorithms. The population size “*N*” and the number of iterations “i” in the AHA framework are factors, which influence the algorithm's temporal complexity. As a type of optimization problem, the feature selection problem may be improved by reducing the number of training features in classification accuracy, and the dimensionality D of the samples also affects the efficiency of the algorithm throughout the optimization process. Table 1 shows the calculated mean and standard deviation results with different dataset for optimization of proposed algorithm.

Figure 4 shows the performance analysis of various algorithms for comparison with mean value. The accuracy of the mean value is always high in the proposed algorithm AHA with moderate iteration such as 50 numbers.

ISSN: 2582-337X
Submitted: 26.05.2021
Revised: 29.06.2021
Accepted: 19.07.2021
Published: 23.07.2021

Table 1. Mean and Standard Deviation Results of Different Datasets

DATASET	GA		AHA(Low)		AHA(Mod)		Wolf optimization algorithm	
	Mean	STD	Mean	STD	Mean	STD	Mean	STD
ANDROID	0.891	0.006	0.900	0.004	0.942	0.003	0.923	0.008
SONAR	0.834	0.019	0.891	0.011	0.900	0.017	0.725	0.016
SPAM BASE	0.861	0.009	0.920	0.002	0.919	0.006	0.909	0.011

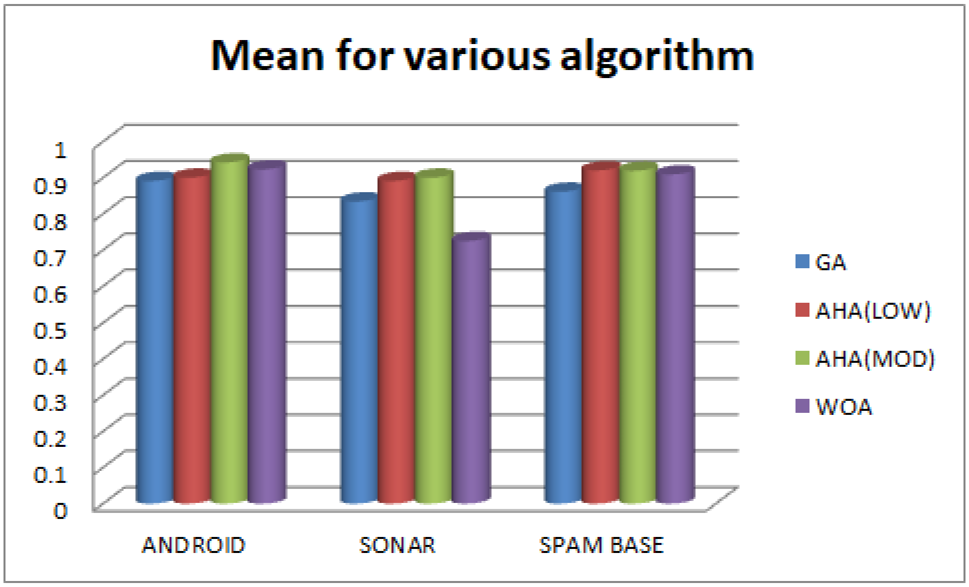

Figure 4. Performance Analysis of Various Algorithms by Mean Value

In summary, the time complexity of the technique relies on the population and the main factors that influence computer complexity are the dimensions. The AHA alters the repetitious process of the traditional genetic algorithm and increases the diversity of populations by changing

ISSN: 2582-337X
Submitted: 26.05.2021
Revised: 29.06.2021
Accepted: 19.07.2021
Published: 23.07.2021

population size [27]. The algorithm's performance should not be exaggerated by mutation actions. Figure 5 show the performance analysis of our proposed algorithm by standard deviation value.

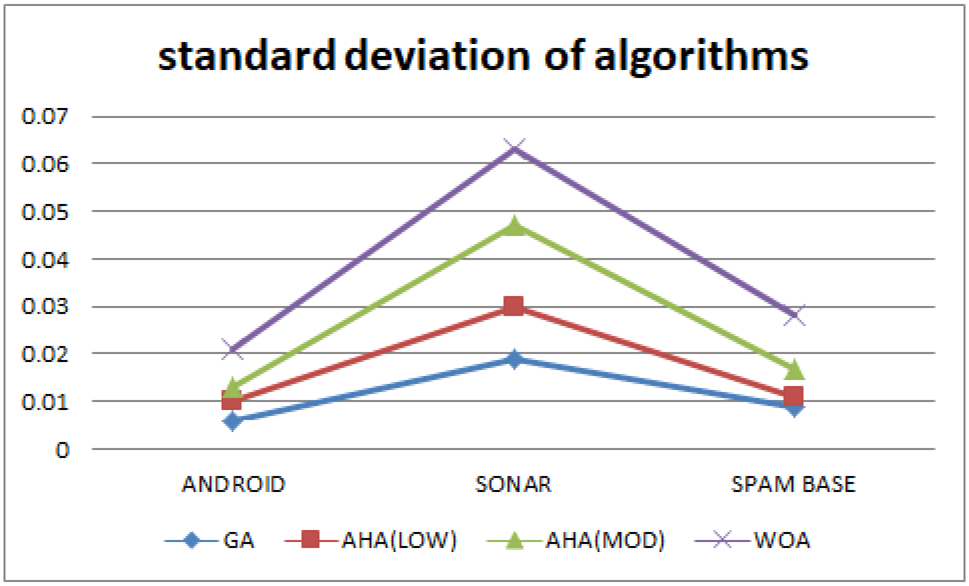

Figure 5. Performance Analysis of Various Algorithms by Standard Deviation

In this section, the influence of various probability of transmission on the algorithm was verified first and then the efficiency of the existing and proposed malware detection techniques were compared. All outcomes are mean results with minimal independent run iterations.

6. Conclusion

The advantages of an adaptive hybrid approach to the evolutionary algorithm have been shown in this research paper for addressing the theoretical selection of issues. The effectiveness has computed in terms of accuracy in mean and standard deviation value of overall our proposed framework. The algorithm is created and investigated by altering the population size and rate of mutation to balance the algorithm to achieve the optimum variable rate to address the issue of this kind. The algorithm is finally utilized. We need to understand the drawbacks of our proposed

ISSN: 2582-337X
Submitted: 26.05.2021
Revised: 29.06.2021
Accepted: 19.07.2021
Published: 23.07.2021

adaptive hybrid method algorithm. The method described in this article is designed to solve the specific problem of optimizing functionality selection. The redundancy of the Crossover operator is specific for this optimization problem category. Secondly, the selection problem for a data set is optimized, and the number of samples and the dimension of data is the main factors influencing runtime or the method called runtime. For future study, it is thus an essential job to use the method for large cases without adding computing complexity.

References

[1] Ranganathan, G. "Real time anomaly detection techniques using pyspark frame work." Journal of Artificial Intelligence 2, no. 01 (2020): 20-30.

[2] Jose, Rinu Rani, and A. Salim. "Integrated static analysis for malware variants detection." In International Conference on Inventive Computation Technologies, pp. 622-629. Springer, Cham, 2019.

[3] Sivaganesan, D. "A Data Driven Trust Mechanism Based on Blockchain in IoT Sensor Networks for Detection and Mitigation of Attacks." Journal of trends in Computer Science and Smart technology (TCSST) 3, no. 01 (2021): 59-69.

[4] Kumar, Ashwin A., G. P. Anoosh, M. S. Abhishek, and C. Shraddha. "An Effective Machine Learning-Based File Malware Detection—A Survey." In International Conference on Communication, Computing and Electronics Systems, pp. 355-360. Springer, Singapore, 2020.

[5] Adam, Edriss Eisa Babikir. "Evaluation of Fingerprint Liveness Detection by Machine Learning Approach-A Systematic View." Journal of ISMAC 3, no. 01 (2021): 16-30.

[6] Suma, V. "Community Based Network Reconstruction for an Evolutionary Algorithm Framework." Journal of Artificial Intelligence 3, no. 01 (2021): 53-61.

ISSN: 2582-337X
Submitted: 26.05.2021
Revised: 29.06.2021
Accepted: 19.07.2021
Published: 23.07.2021

[7] Soni, Jayesh, Suresh K. Peddoju, Nagarajan Prabakar, and Himanshu Upadhyay. "Comparative Analysis of LSTM, One-Class SVM, and PCA to Monitor Real-Time Malware Threats Using System Call Sequences and Virtual Machine Introspection." In International Conference on Communication, Computing and Electronics Systems: Proceedings of ICCCES 2020, vol. 733, p. 113. Springer Nature, 2021.

[8] Hamdan, Yasir Babiker. "Faultless Decision Making for False Information in Online: A Systematic Approach." Journal of Soft Computing Paradigm (JSCP) 2, no. 04 (2020): 226-235

[9] Agrawal, Prerna, and Bhushan Trivedi. "AndroHealthCheck: A Malware Detection System for Android Using Machine Learning." In Computer Networks, Big Data and IoT, pp. 35-41. Springer, Singapore, 2021.

[10] Chen, Joy Iong Zong, and Lu-Tsou Yeh. "Graphene based Web Framework for Energy Efficient IoT Applications." Journal of Information Technology 3, no. 01 (2021): 18-28.

[11] Smys, S., and Haoxiang Wang. "Security Enhancement in Smart Vehicle Using Blockchain-based Architectural Framework." Journal of Artificial Intelligence 3, no. 02 (2021): 90-100.

[12] Raj, Jennifer S. "Security Enhanced Blockchain based Unmanned Aerial Vehicle Health Monitoring System." Journal of ISMAC 3, no. 02 (2021): 121-131.

[13] Yang Huan, Zhang Yuqing, Hu Yupu, etc. Android application malicious behavior detection system based on multiple characteristics; Chinese Journal of Computers, 2014, 37(1):15-27.

[14] Amos B, Turner H, White J. Applying machine learning classifiers to dynamic Android malware detection at scale. 2013.

[15] Zhang B T. Hypernetworks: A Molecular Evolutionary Architecture for Cognitive Learning and Memory. IEEE Computational Intelligence Magazine, 2008, 3(3):49-63.

[16] Yerima, Suleiman Y., Sakir Sezer, and Gavin McWilliams. "Analysis of Bayesian classification-based approaches for Android malware detection." IET Information Security 8, no. 1 (2014): 25-36.

ISSN: 2582-337X
Submitted: 26.05.2021
Revised: 29.06.2021
Accepted: 19.07.2021
Published: 23.07.2021

[17] Shabtai, Asaf, Yuval Fledel, and Yuval Elovici. "Automated static code analysis for classifying android applications using machine learning." In 2010 international conference on computational intelligence and security, pp. 329-333. IEEE, 2010.

[18] Suma, V., and Wang Haoxiang. "Optimal Key Handover Management for Enhancing Security in Mobile Network." Journal of trends in Computer Science and Smart technology (TCSST) 2, no. 04 (2020): 181-187.

[19] Sanz, Borja, Igor Santos, Carlos Laorden, Xabier Ugarte-Pedrero, Pablo Garcia Bringas, and Gonzalo Álvarez. "Puma: Permission usage to detect malware in android." In International Joint Conference CISIS'12-ICEUTE 12-SOCO 12 Special Sessions, pp. 289-298. Springer, Berlin, Heidelberg, 2013.

[20] Peiravian, Naser, and Xingquan Zhu. "Machine learning for android malware detection using permission and api calls." In 2013 IEEE 25th international conference on tools with artificial intelligence, pp. 300-305. IEEE, 2013.

[21] Yildiz, Oktay, and Ibrahim Alper Doğru. "Permission-based android malware detection system using feature selection with genetic algorithm." International Journal of Software Engineering and Knowledge Engineering 29, no. 02 (2019): 245-262.

[22] Jung, Jaemin, Kyeonghwan Lim, Byoungchul Kim, Seong-je Cho, Sangchul Han, and Kyoungwon Suh. "Detecting malicious android apps using the popularity and relations of apis." In 2019 IEEE Second International Conference on Artificial Intelligence and Knowledge Engineering (AIKE), pp. 309-312. IEEE, 2019.

[23] Simoes, A.; Costa, E. Using genetic algorithms with sexual or asexual transposition: a comparative study. Proc. CEC00 **2000**, 10, 1196–1203.

[24] Farasat, A.; Menhaj, M.B.; Mansouri, T.; Moghadam, M.R. ARO: A new model-free optimization algorithm inspired from asexual reproduction. Appl. Soft Comput. **2010**, 10, 1284–1292.

[25] Sharma, Kapil, Anish Singh, and Prateek Arora. "A Study of Android Malware Detection Using Static Analysis." In Computer Networks and Inventive Communication Technologies, pp. 1071-1080. Springer, Singapore, 2021.

[26] Komatwar, Rupali, and Manesh Kokare. "Malware Identification and Classification by Imagining Executable." In Proceedings of International Conference on Intelligent Computing, Information and Control Systems, pp. 375-387. Springer, Singapore, 2021.

[27] Shakya, Subarna. "IoT based F-RAN Architecture using Cloud and Edge Detection System." Journal of ISMAC 3, no. 01 (2021): 31-39.

Author's Biography

B Vivekanadam is a senior lecturer in the Department of Computer Science and Multimedia at Lincoln University College, in Malaysia. His major area of research are machine learning, neural network algorithms, image processing, video and signal processing, cloud computing, deep learning, artificial intelligence, object recognition, complex feature extraction and vision graphics.