https://dx.doi.org/10.4314/dujopas.v10i3a.17
ISSN (Print): 2476-8316
ISSN (Online): 2635-3490
Dutse Journal of Pure and Applied Sciences (DUJOPAS), *Vol. 10 No. 3a September 2024*

Framework for the Detection and Classification of Malware using Machine Learning

[1*]Funmilayo Jumoke Akinshola-Awe, [1]A.A. Obiniyi,
[1]Gilbert I.O. Aimufua, [1]Tochukwu Kene Anyachebelu

[1]Computer Science Department,
Nasarawa State University
Keffi
Nigeria.

Email: emilade10@gmail.com

Abstract

Malware constitute a major threat to Network Infrastructure which are vulnerable to several devastating Malware attacks such as Virus and Ransomware. Traditional Antimalware software provides limited efficiency against Malware removal due to evolving evasion techniques capabilities of Malware such as polymorphism. Antimalware only removes Malware they have signatures for and are ineffective and helpless against zero day attack, several research works have made use of supervised and unsupervised learning algorithms to detect and classify Malware but False Positives prevails. This research made use of Machine Learning to detect and classify Malware by employing Machine Learning techniques including Feature Selection techniques as well as Grid Search hyperparameter optimization. Principal Component Analysis was combined with Chi Square to cure the curse of dimensionality. Support Vector Machine, K Nearest Neighbor and Decision Tree were used to train the model separately with two datasets. The research model was evaluated with Confusion Matrix, Precision, Recall and F1 Score. Accuracy of 99%,98.64% and 100% was achieved with K Nearest Neighbor, Decision Tree and Support Vector Machine respectively using CICMalmem dataset which has equal number of Malware and Benign files, K Nearest Neighbor achieved no False Positive. Accuracy of 97.7%,70% and 96% was achieved with K Nearest Neighbor, Decision Tree and Support Vector Machine respectively with Dataset_Malware.csv dataset, K Nearest Neighbor achieved False Positives of 38.The Model was trained separately with default hyperparameters of the chosen algorithms as well as the optimal hyperparameters obtained from Grid Search and it was discovered that optimizing hyperparameters and combining features obtained with Principal Component Analysis and Chi Square to train the Model using the dataset with equal number of Benign and Malicious files(CICMalmem dataset) yielded optimal performance with Support Vector Machine. Future works includes employing deep learning and ensemble learning as classifiers as well as implementing other hyperparameter optimization techniques.

Keywords: Malware Detection, Feature Selection, Hyperparameter Tuning, Grid Search, Machine Learning.

INTRODUCTION

There is rise in the use of Internet which is a global network of interconnected computer networks has brought up new risks and vulnerabilities. One of the main problems facing cybersecurity is malicious attack (Abiola & Marhusin,2018). Malicious software, also referred to as Malware, is intrusive software that is designed with the specific goal to harm, gain

Author for Correspondence

unauthorized access to, or disrupt computer systems. Malware can be in form of virus, worm, adware, spyware, ransomware, and other various forms, each with unique characteristics and modes of operation (Baur, 2003).

These dangerous programs have the ability to infiltrate mobile devices, computers, and networks, compromising personal data, interfering with daily business operations, and resulting in large financial losses. As technology develops, Malware becomes more complex, posing a constant and evolving threat to the Internet. Classifying Malware is one of the most important parts of handling it. The process of categorizing a particular Malware sample into a particular Malware family is known as Malware classification (Helwitt, 2022).

Malware within the same family often shares similar properties, such as behavior, code patterns, or structural characteristics, which can be used to develop signatures for detection and classification purposes. Signatures, which can either be static (based on the binary code) or dynamic (based on runtime behavior), play a crucial role in identifying and categorizing Malware (Walenstein & Lakhotia, A. kjyu2007).

The traditional approach to Malware detection and classification relied heavily on signature-based methods (Helwitt, 2022). Anti-Malware software would compare incoming data against a library of known Malware signature, if a match is found, the software would flag the file or code as malicious. While this approach provided a level of protection against known Malware threats, it struggled to handle emerging and unknown variants, commonly referred to as zero-day attacks (Kwon, Son & Ryu, 2022). The creation of signatures for classification and detection can be facilitated by patterns, or structural traits. The conventional approach to Malware detection and classification relied heavily on signatures, which can be either static (based on the binary code) or dynamic (based on runtime behavior) (Walenstein & Lakhotia, A. kjyu2007). Incoming data would be compared to a database of known Malware signatures by anti-Malware software, which would mark the file or code as harmful if a match was found. While this method provided some protection against known Malware threats, it was unable to handle newly emerging and unknown versions, or zero-day attacks. The huge number of polymorphic and metamorphic Malware which change their code patterns or behavior to evade detection makes relying solely on signature-based methods inadequate. To address the limitations of traditional approaches, researchers and cybersecurity professionals turned to machine learning techniques. Machine learning leverages algorithms and statistical models to analyze and identify patterns in large datasets (Javaheri et al., 2018). By training models on a vast amount of labeled Malware samples, machine learning algorithms can learn to recognize malicious patterns and classify unknown samples based on their similarities to known Malware families. The use of machine learning algorithms for Malware detection and classification have shown promising results (Saad et al., 2019). An enormous increase in Malware attacks has resulted from the growing usage of the Internet and the advent of digital technology, posing a serious risk to people, companies, and vital infrastructure. Because Malware is dynamic and complicated, it is difficult to identify and categorize using conventional signature-based methods, particularly in light of the appearance of polymorphic and metamorphic variants.

Since antimalware software can only detect Malware for which it has signatures, it cannot detect zero-day attacks. Furthermore, updating new and evolving signatures can be time- and resource-consuming. Thus, there is a pressing need to design an effective framework for Malware detection and classification using machine learning. A comprehensive framework is developed using the vast number of tagged Malware samples to identify patterns, behaviors, and characteristics that distinguish one Malware family from another. However, the efficacy

of currently available related efforts in identifying Malware and preventing zero-day attacks is still limited due to false positives and false negatives.

Establishing a trustworthy machine learning-based framework for Malware detection and classification requires overcoming several challenges. The framework has to be scalable, flexible, and real-time performing to satisfy the demands of dynamic and evolving Malware variants. Choosing appropriate machine learning models and algorithms, creating effective feature extraction and selection strategies, managing imbalanced datasets, interacting with high-dimensional and heterogeneous data sources and prevailing false positives and false negatives are some of the shortcomings in related studies.

Furthermore, few publications have attempted to optimize classifiers or dealt with insufficient training data sets (Di Troia, 2021). , Malware samples from distinct families are categorized using a variety of features (Enisa, 2021). One of the simplest yet most important categorization methods in machine learning is K-Nearest Neighbors. It is used in pattern recognition, data mining, and intrusion detection and is a member of the supervised learning domain (Phyu, 2009).

Machine learning algorithms was used in the context of advanced Malware detection of highly obfuscated files, the models was developed to combat the complex challenge of detecting zero-day attack with SVM and Clustering algorithms achieving reasonable accuracy. Experiments were also introduced to text generic Malware model where SVM was able to obtain promising result while KNN and Random Forests proved to be more effective in detecting Obfuscated Malware (Di Troia, 2021).

Two level classifiers were used to construct a framework for identifying and classifying different files (exes, pdfs, etc.) as benign or harmful: macro for Malware detection and micro for malicious file classification (such as Trojan, Spyware, etc.). Random Forest Tree, J48 Decision Tree, and SMO algorithms were used to train the model,J48 Decision Tree outperformed other classifiers in terms of accuracy and performance. However, not all of the attributes may have been extracted, making the analysis to be skewed because only 220 samples were used. The research executed the sample files in the virtual environment using Cuckoo Sandbox, which produced static and dynamic analysis reports. (Sethi et al., 2017).

Hossai (2020) developed a model that enhanced accuracy by optimizing the hyperparameters of twenty classifiers belonging to nine machine learning families. Sixteen out of the twenty classifiers which included Support Vector Machine performed better with optimized hyperparameters when compared to the accuracy of the model when the default hyperparameters were used to train the model. The dataset employed for training is large enough to contain many Malware families.

API, opcodes, n-grams control flow graphs and Dynamic Link Libraries, strings, function length and function length frequency are some of the vectors explored to analyse and detect Malware. A collection of 10, 072 unique samples was classified into 14 Malware families. The model was trained with Support Vector machine (SVM) and managed to classify 88% of the provided testing binaries to their correct Malware family (Rieck et al., 200

Hyperparameters were defined as regulated parameters that are selected for training a model that control the training process itself, a model was proposed in which the hyperparameters of Random Forest were tunned to achieve higher accuracy for Birds Species Identification

System. The higher the estimators, the better the performance of the model but the computational cost becomes higher with more time of execution.(Ganasan et al., 2022)

MALWD&C model was proposed by Buriro et al.(2022) where BODMAS dataset was used to train the model that was able to detect Malware with accuracy of 99.56% with Random forest. Combination of features obtained by Principal Components Analysis and Chi2 with hyperparameter optimization was tested to detect Diabetes effectively employing logistic tree classifier giving higher accuracy than when features obtained with PCA or Chi2 were separately used to test the model. The fusion of PCA and Chi2 feature selection technique is one of the concepts employed in this research.(Rupapara et al., 2023)

METHODOLOGY

The goal of the research is to accurately detect and classify Malware and address the issue of low dataset size that cannot adequately generalize findings and accurately detect new Malware variant with low false positives and false negatives. The classifiers employed in the development of the model have several hyperparameters out of which few were selected for optimization (Hossain & Ayub, 2020).The approach is to reduce the dimensions of the two datasets chosen for this research by combining features obtained from Principal Components Analysis and Chi Square with which the model is trained with optimized hyperparameters of KNN, Decision Tree and Support Vector Machine.

The model was finally evaluated using standard performance metrics. Features obtained from the implementation of Principal Component Analysis and Chi Square were combined to boost accuracy of detection but the dataset has to be standardized using Standard Scaler because the variables in the datasets have different scales(Rupapara et al., 2023) .The issue of null Values was addressed to ensure a clean and noiseless data which can affect the performance of the model negatively. The datasets were split into 80% training data and 20% test data. The training data was used to train the model while the test data was used to make prediction. Datasets are usually split into train and test data to avoid overfitting which is an instance where Machine Learning models fits its training data and fails to fit additional data. K fold cross validation was carried along with the Grid Search to validate data and reduce overfitting (Sharma et al., 2021).

This research model is illustrated in Figure A

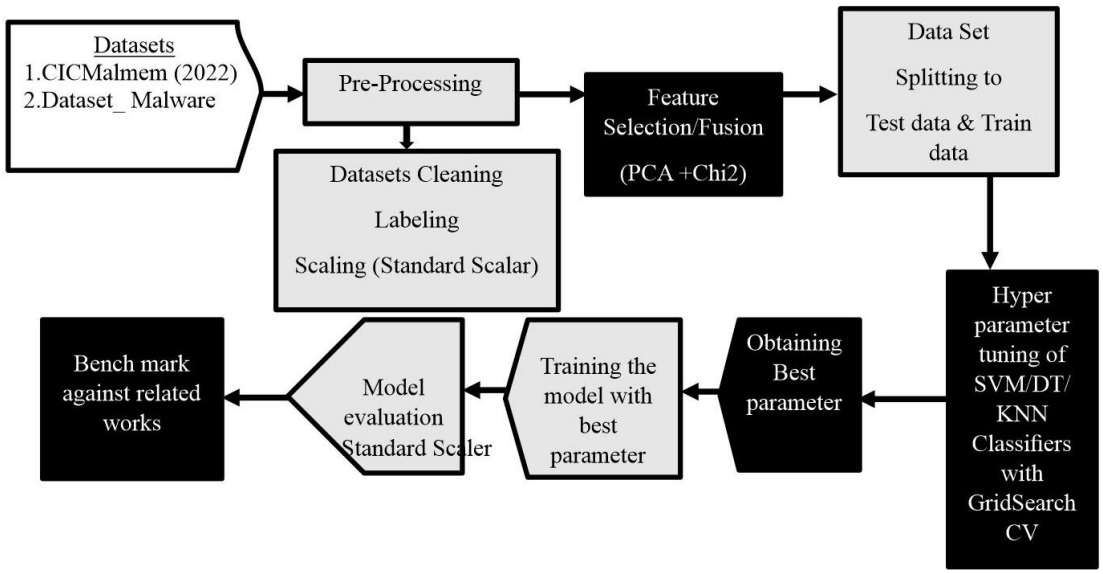


Figure A Research Model

Dataset **Sample**
Two Datasets obtained from Kaggle were used for the study.

The datasets description
Two datasets were obtained from the Kaggle which were used to test and train the models using machine learning. The datasets used for training will be 80% of the dataset while the remaining 20% will be used for testing. Sizeable datasets containing several Malware Families were used for training and testing the model. The data was validated during the hyperparameter optimization using K fold cross validation techniques. This is very important to avoid overfitting.

a) CIC Malmem 2022

This Dataset contains obfuscated Malware and was designed to detect obfuscated Malware detection methods through the memory.

The Dataset was created by the Canadian Institute for Cybersecurity based at the University of New Brunswick. The Dataset is balanced with it being made up of 50% Malware and 50% Benign Memory Dumps. The Database contains a total of 58,596 records with29,298 Malicious and 29,298 Benign files. The Database size is 18.98MB with 57 dimensions corresponding to the features existing in the database with 58,596 rows.
This dataset can be imported via pandas into Python from Kaggle.com website and its illustrated in Figure B. The dataset is made up of fifteen Malware families out of which five families are Trojans which made up 16.2%of the dataset, five families are Spyware which made up 17.1% of the dataset and the remaining 16.7% of the datasets contains five families of Ransomware. The dataset is illustrated in Figure B

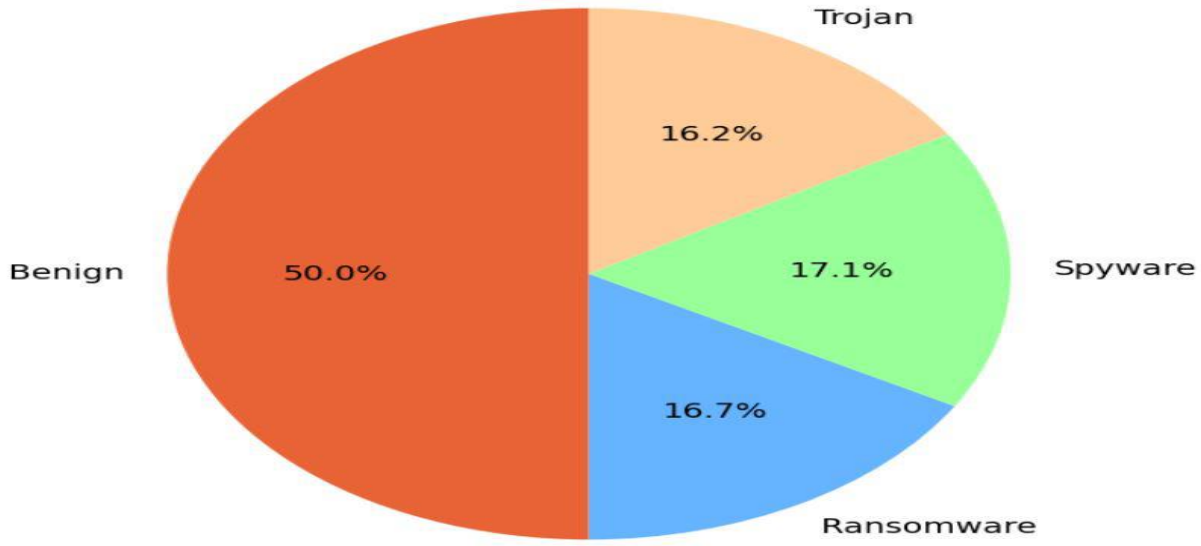


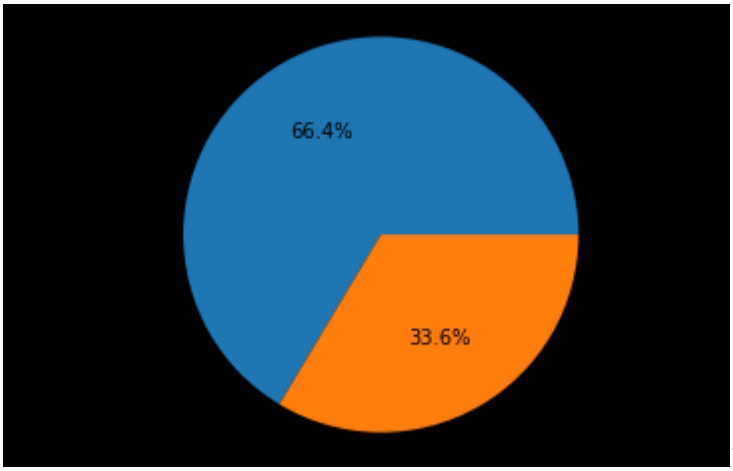
Figure B: CIC Malmem 2022 Complete dataset breakdown(unb.ca)

b) Dataset_Malware.csv

This dataset was created by Mai Daly. It was built using a Python Library and contains benign and malicious data from Portable Executable (PE) Files and uploaded to Kaggle website. The file consists of total 19611 samples out of which 14599 are VirusShare Malware and were classified as malicious files while the remaining 5,012 are Benign. The aim of the dataset is to detect and classify a Malware using a machine learning algorithm. The file size is 6.72 MB with up to 75 dimensions which corresponds to the number of attributes/features existing in the dataset.

This dataset can be imported via pandas and loaded into Python. The loaded dataset appeared as seen in Figure C

Malware



Benign

Figure C: Dataset_Malware.csv ((Jummani et al., 2022)

Three training models will be applied on the datasets namely Decision Tree, K Nearest Neighbor and SVM Classifier.

With this dataset, several classifiers such as Support Vector Machine achieved Accuracy of 96.039% in Malware detection. (Jummani et al., 2022).

Evaluation of the Model

This involves evaluating the trained model for its performance using Accuracy and Confusion Metrics. .

Accuracy – this simply measures how often the model correctly makes predictions

TP (True Positive) – Successful identification of an attack by the Model, occurs when an attack is predicted and its true

TN (True Negative): The model did not forecast any attack, and this is accurate.

FP (False Positive): The model anticipated an attack, but it is untrue.

FN (False Negative): The model did not forecast any attack, and this is untrue.

The confusion Matrix is illustrated D

		Predicted Class	
		Normal	Attack
Actual Class	Normal	True Negative (TN)	False Positive (FP)
	Attack	False Negative (FN)	True Positive (TP)

Figure D Confusion Matrix (Suresh,2020)

Hyperparameters

Illustrated in Table 1 are the common features of the selected classifiers for the Research work as well as their corresponding Hyperparameters

Table 1 Classifiers Features and Hyperparameters

Classifier Features	Hyperparameters
Decision Tree Robustness to noise Fast runtime Robust predictors	• Min samples split (minimum number of samples required to split an internal mode (the default is 2). • Criterion - measure of the quality of tree/labels on a mode (gini or entropy). • Max depth (maximum depth of the tree, default = none)
KNN Simple to implement, flexible to multiple features and classifies well in practice with enough data representation.	• N_neighbors (number of neighbors (default = 5) • Weight contribution of members of the neighborhood via different weight (uniform or distance. • Metric (Euclidean, Manhattan or Mikowski)
SVM Can solve complex problem with appropriate kernel function It scales relatively well to high dimensional data has high accuracy of prdiction	• Kernel (choice of kernel that will control the manner in which the input variables will be projected) = Linear/Poly/rbf/sigmoid .

The model was trained with Decision Tree, Support Vector Machine and K Nearest Neighbor's selected hyperparameters as illustrated in Table 2

Table 2: Hyper Parameter Tuning

Classifier	HYPER PARAMETER
KNN	Default parameter Weight = uniform , metric option = minkowski , k=5 Grid parameters K=1-30 CV = 5 Weight =uniform , distance Metric = Euclidean, Manhattan, Minkowski
SVM	Default parameters c=1.0, kernel =rbf, gamma=scale Grid parameter C = [0.1,1,10] Kernel = [linear, rbf ,poly, sigmoid] Gamma:[0.001,0.01,0.1,1]
DT	DEFAULT PARAMETERS Criterion=gin, max_ depth=none, Minimum sample split=2 Minimum sample leaf =1 Grid parameter Criterion =[Gini, entropy] Max_depth=[none,5,10,15] Min_sample_split =[2,5,10] Min_sample_leaf =[1,2,4]

RESULTS AND DISCUSSION

Illustrated in Table 3 is the performance of the research model compared to models in related works using the two datasets employed in this study. Dataset A is CICMalmem (2022) and Dataset B is Dataset_Malware. Optimal result with false positive of zero was achieved by the research model with KNN and false negative of zero was achieved with SVM. Accuracy of 100% was achieved with only SVM when its hyperparameters were optimized. The findings are compared to the performance obtained in related models.

Table 3 Benchmarking Against Related works

S/N	Research	Classifier	Accuracy (%)	FP	FN	TP	TN
1.	Research Model (Default Hyper Parameters with PCA and Chi2combination) Dataset A	KNN DT SVM	99 97 99	0 300 2	1 0 0	5882 5626 5900	5837 5794 5818
2.	Research Model (Hyper parameter Optimization using Grid SearchCV with PCA and Chi2 Combination)Dataset A	KNN DT SVM	99 98.64 100	0 157 1	1 2 0	5810 5598 5765	5909 5963 5954
3.	Immune-Based System to Enhance Malware Detection.(Jerbi et al., 2023)	KNN DT SVM	70.47 71.03 95.35				
4.	Supervised and unsupervised learning techniques utilizing Malware datasets.(Smith et al., 2023)	KNN DT	99.91 99.99				
5.	Malware detection using memory analysis data in big data environment. (Dener et al., 2022)	DT SVM	99.79 99.14	3			
6.	Research Model using hyper parameter optimization with Grid Search and PCA and Chi2 Combination (Dataset B)	KNN DT SVM	97.7 76 95.84	38 780 138	52 398 25	2826 2538 2871	1007 216 889
7.	Research Model using hyper parameter optimization with Grid Search and PCA and Chi2 Combination (Dataset B)	KNN DT SVM	97.7 76 95.84	38 780 138	52 398 25	2826 2538 2871	1007 216 889
8.	A comparative analysis of Malware anomaly detection.(Sharma et al., 2021)	DT SVM	40.32 96.09				
9.	A Supervised Machine Learning Algorithm for Detecting Malware. (Ayeni, 2022)	DT KNN	97.77 96.33	477 459	186 580	14648 14648	14689 14295
10.	Effective One-Class Classifier Model for Memory Dump Malware Detection.(Al-Qudah et al., 2023)	SVM(OCSVM) One class SVM (Using) PCA (occ-PCA	99.4%				

The research model was able to outperform most of the high performing models serving as benchmark with high accuracy and no false positive and false negative, in addition, the datasets employed in the research have sufficient Malware families. Dataset A with higher dimension and equal number of Malware and Benign files performed better in Malware detection.

CONCLUSION

Detection and classification of Malware with precision and maximal accuracy is highly essential in business to preserve sensitive information which are daily exposed and vulnerable to zero day attack of new Malware variants with unknown signatures. This study has been able to develop a Model that was able to predict Malware with accuracy of 100% with SVM and reduce False Positives and False Negatives to 1 and 0 respectively using balanced dataset with equal number of Benign and Malicious files (CICMalmem). The Model can be explored in the development of Antimalware.

In conclusion, the research work has been able to detect Malware with high Accuracy by optimizing the hyperparameters of the chosen classifiers and reducing the dimensions of datasets although the scope of work did not include classification of Malware into their corresponding families. Deep learning (Neural Networks) can be explored to detect Malware as well as other Feature Selection techniques.
Bayesian Optimization techniques and Random Search method can also be implemented to detect optimal hyperparameters of the chosen Classifiers.

REFERENCES

Al-Qudah, M., Ashi, Z., Alnabhan, M., & Abu Al-Haija, Q. (2023). Effective One-Class Classifier Model for Memory Dump Malware Detection. *Journal of Sensor and Actuator Networks*, *12*(1). https://doi.org/10.3390/jsan12010005

Ayeni, O. A. (2022). A Supervised Machine Learning Algorithm for Detecting Malware. *Journal of Internet Technology and Secured Transactions*, *10*(1), 764–769. https://doi.org/10.20533/jitst.2046.3723.2022.0094

Dener, M., Ok, G., & Orman, A. (2022). Malware Detection Using Memory Analysis Data in Big Data Environment. *Applied Sciences (Switzerland)*, *12*(17). https://doi.org/10.3390/app12178604

Ganasan, J., Hashim, A. S., & Ibrahim, N. (2022). Lecture Notes in Networks and Systems 279. In *Software Engineering Perspectives in Systems* (Vol. 501, Issue Iciil). https://link.springer.com/bookseries/15179%0Ahttp://www.springer.com/series/15179

Hossain, S. M., & Ayub, M. A. (2020). Parameter Optimization of Classification Techniques for PDF based Malware Detection. *ICCIT 2020 - 23rd International Conference on Computer and Information Technology, Proceedings*, 19–21. https://doi.org/10.1109/ICCIT51783.2020.9392685

Jerbi, M., Dagdia, Z. C., Bechikh, S., & Said, L. Ben. (2023). Immune-Based System to Enhance Malware Detection. *2023 IEEE Congress on Evolutionary Computation, CEC 2023*. https://doi.org/10.1109/CEC53210.2023.10254159

Jummani, F., Chaudhari, S., & Gujar, S. (2022). *COMPARATIVE ANALYSIS OF MALWARE DETECTION DATASETS USING DIFFERENT MACHINE LEARNING CLASSIFIERS Result Analysis for Accuracy in ML Classifiers Techniques*. *9*(January), d698–d703.

Louk, M. H. L., & Tama, B. A. (2022). Tree-Based Classifier Ensembles for PE Malware Analysis: A Performance Revisit. *Algorithms*, *15*(9), 1–15. https://doi.org/10.3390/a15090332

Rupapara, V., Rustam, F., Ishaq, A., Lee, E., & Ashraf, I. (2023). Chi-Square and PCA Based

Feature Selection for Diabetes Detection with Ensemble Classifier. *Intelligent Automation and Soft Computing*, *36*(2), 1931–1949. https://doi.org/10.32604/iasc.2023.028257

Sharma, P., Chaudhary, K., Wagner, M., & Khan, M. G. M. (2021). A Comparative Analysis of Malware Anomaly Detection. In *Advances in Intelligent Systems and Computing* (Vol. 1158, pp. 35–44). https://doi.org/10.1007/978-981-15-4409-5_3

Smith, D., Khorsandroo, S., & Roy, K. (2023). Supervised and Unsupervised Learning Techniques Utilizing Malware Datasets. *2023 IEEE 2nd International Conference on AI in Cybersecurity, ICAIC 2023*. https://doi.org/10.1109/ICAIC57335.2023.10044169