

Artificial Intelligence Powered Machine Learning Approach for effective Malware Detection and Enhancing accuracy in Personal Data Privacy

Apurva Mishra¹, Rajiv Bhalla²

Research Scholar¹, Professor²

Chandigarh University, NH-05 Chandigarh-Ludhiana Highway, Mohali, PB, India

Corresponding Author: apurvamishra2141@gmail.com

Abstract— In the world of computers and software, data is an asset; a personal, valuable, and important component in every person's life. Malware used to extract hidden and personal data is slowly spreading around the globe. Ordinary signature-based malware discovery procedures are finding it troublesome to keep up with the consistent development of malware. A viable substitute is machine learning (ML), which can recognize intricate patterns and adjust to new threats. This study compares 11 different Machine Learning models, to find the best model that can detect malware. The algorithms are trained and tested on a dataset called TUNADROMD with and without using Principal Component Analysis (PCA). The Random Forest is found to be the best-performing model with and without PCA with accuracies of 0.99 and 0.98 respectively. The worst-performing model is found to be Logistic Regression both with and without PCA with accuracies of 0.98 and 0.81 respectively.

Keywords— Malware, TUNADROMD, Principal Component Analysis, Random Forest, Logistic Regression

1. INTRODUCTION

Malicious software designed to exploit the confidentiality of personal data, pick up unauthorized get to other computer frameworks and systems, or indeed manipulate or harm the integrity and availability of data is widely known as Malware. Malware continues to grow in number and strength every year and is becoming a potential threat to global cyber security [1]. AV-Test Institute conducted a study revealing that approximately 350,000 new hazardous applications and programs are reported every day. In 2020, about 897 million malicious codes were documented in malware statistics [2]. Malware has developed to the extent of evading many anti-malware or anti-virus tools. Static malware analysis is no longer helpful in preventing malware attacks. Multiple researches are, thus, being conducted to develop alternative approaches for malware detection and elimination [3].

Until now, three effective methods for detecting and analysing Malware have been

developed namely Static analysis, Memory analysis, and Energetic examination. Inactive Malware examination could be a strategy of recognizing destructive program without executing it. Memory analysis is a method in which the Malware is analysed after it has been executed. Dynamic

Malware analysis is another method in which the activity of the Malware is observed by executing the Malware [2]. Programmed energetic malware examination when coupled with information mining assignments, such as machine learning, can be utilized to accomplish productivity and viability in malware location [3].

The involvement of Machine Learning within the location and cancellation of Malware has given birth to a completely new prospect for research in this field. Machine Learning is currently a rising tool in all fields. It is being used in medical as well as technical fields. Some of the successful projects in the medical field are the diagnosis of mental disarranges such as Alzheimer's infection, Parkinson's malady, and sadness [4-6] and the diagnosis of many physical disorders such as malaria, heart disease, and hepatitis [7-9]. On the other hand, utilize of Machine Learning within the specialized field is found in fraud detection, recommendation systems, and software fault prediction [10-12].

In this paper, distinctive machine-learning models have been executed on a dataset called TUNADROMD. The machine learning models foresee on the off chance that a given occasion is malware depending on the properties. There are 9 ML models executed in this paper to be specific Bolster Vector Machine, Arbitrary Timberland, K-Nearest Neighbors, Calculated Relapse, Multi-Layer Perceptron, Choice tree, AdaBoost, Angle Boosting, and XGBoost. The models have been actualized both with and without Guideline Component Investigation. The comes about are at last compared to discover the best-fit show for the discovery of malware. The rest of the segments are II. Related Work, III. Fabric and Strategy, IV. Technologies used V. Experimentation and Results, VI. Conclusion and Future Work.

2. RELATED WORK

Malware is slowly becoming the world's biggest threat to cyber security. Ever since it started evading anti-malware tools, researchers have been trying to develop better methods to fight against malware.

Hemant Rathore et al. (2018) [13] utilized Opcode recurrence and connected unsupervised and administered machine learning and profound learning for the classification of Malware. In this paper, different machine learning models and profound learning models are utilized to identify Malware. The comes about appear that Random Forest is the best-performing machine learning demonstrate with an exactness of 99.78% and it too outflanks Profound Neural Network (DNN) which is found to be the leading profound learning show with an exactness of 99.70%.

Jin Li et al. (2018) [14] introduced a three-level pruning method to find the foremost noteworthy consents that can successfully recognize between kind and malevolent applications. This paper brings forward Critical Permission Identification (SigPID) which

may be a framework that employs authorizations to distinguish malware. After classifying malware as generous or noxious, SigPID employs ML classification to classify it into the diverse families of malware. The results appeared that the Back-Vector Machine accomplished 90% accuracy in recognizing malware and kind applications. SigPID was found to identify 93.62% of malware and 91.4% of obscure or unused malware tests in a given dataset. Chün-Ying Huang et al. (2013) [15] presented a permission-based malware locator. AdaBoost, Naïve Bayes, Choice tree, and Bolster Vector Machine (SVM) are utilized for three different datasets. SVM was found to outperform all other models and the permission-based detector was found to detect more than 81% of malicious samples.

Radu S. Pirscoveanu et al. (2015) [16] address the challenge of automating and scaling the analysis and classification of large amounts of malware. This study has used an extended version of the Cuckoo Sandbox for testing and tracing malware. This approach was tried on a total of 42,000 malware tests. Administered machine learning was utilized for sifting novel malware from the known malware alongside detecting malware. Irregular Woodland classifier was found to have the leading exactness score of 98% in identifying and recognizing malware and its type. Sherif Saad et al. (2019) [17] conversation approximately the issues that can constrain the victory of machine-learning procedures in malware detection. The paper defines three such issues and also presents the possible solutions to the arising problems. They also describe the requirements that foolproof malware detection may require. Ritesh Sharma et al. (2019) [18] bring forward the security challenges faced in cloud computing and its plausible solutions. The paper describes multiple security issues such as Insider and outsider threats, data leakage, multitancy, and data segregation. It discusses the solution to each problem in a tabular form.

3. MATERIALS AND METHODOLOGY

This SEGMENT describes the dataset, the additional components used, the different ML models and the software and hardware used. The sections are distributed as A. Dataset Used, B. Principal Component Analysis (PCA), C. Data Exploration and Splitting, D. Software and Hardware

A. *Dataset Used*

The dataset used in this paper is called TUNADROMD [19]. It is created by Parthajit Borah, DK Bhattacharyya, and JK Khalita in the year 2021 to support research based on malware and the effectiveness of malware detection methods. TUNADROMD is taken from Kaggle and contains 4465 instances as rows and 241 attributes as columns. The target attribute is a category that shows if an instance is malware or not and is designated as “*Label*”. This dataset is a preprocessed version of TUANDROMD dataset. TUNADROMD has been trained and tested by multiple ml models. The independent variable contains 4465 rows and 240 columns i.e. all columns except Label. The dependent variable contains 4465 rows and 1 column i.e. Labels.

B. *Principle Component Analysis (PCA)*

Each data point of a dataset is scattered in a high- dimensional space where each dimension represents a feature. The dimensions are analyzed and the dimensions holding the most variance are selected to be the Principal Components. The data is then projected onto these new Principal Components. This is the process of Principle Component Examination (PCA). This makes a difference in diminishing the dimensionality of information which reduces the complexity of data and thus, reduces the chances of overfitting. All machine Learning models used in this study are first implemented without PCA and then with PCA.

C. *Data Exploration and Splitting*

Without PCA

The dataset was found to be balanced and to have no NAN values, thus requiring no preprocessing. The dataset was part into preparing and testing factors in a proportion of 7:3 with a test size of 0.3 and random state 101.

With PCA

The dataset was scaled using Standard Scaler. The dataset was then oversampled using ADASYN with a random state of 42. Finally, PCA is applied to the data, and the data is part into preparing and testing factors with a test estimate of 0.3 and an irregular state of 101.

D. *Software and Hardware*

A workstation with an 11th Gen Intel(R) Core (TM) i7- 1165G7 @ 2.80GHz CPU processor, 16 GB RAM, and Windows 11 OS was used for the study. The tools utilized were Scikit-Learn modules and keras, along with Python version 3.11.1 in the Vs Code IDE.

4. TECHNOLOGIES USED

The sub-sections are distributed as A. Back Vector Machine (SVM), B. Random Forest C. Choice Tree, D. K- Closest Neighbor, E. Calculated Relapse, F. Multi-layer Perceptron, G. XGBoost, H. AdaBoost, I. Slope Boosting Machine, J. Light Angle Boosting Machine, K. Back Propagation Neural Network

A. *Support Vector Machine (SVM)*

Back Vector Machine or SVM may be a special machine learning calculation that maximizes its accuracy by selecting the maximum margin that separates the optimal hyperplane. Due to this quality of SVM, it often predicts more accurately than other models. It has thus, become an extremely useful tool in getting accurate predictions, especially in the medical field [20].

B. *Random Forest*

The Irregular Woodland classifier is made up of a few tree classifiers that combine to classify the inputs. It could be a administered machine learning method which can he utilized with both classification and relapse. It is based on the concept of gathering

learning. Bagging is a technique that creates a training dataset by selecting a feature or a combination of features, and then randomly selecting replacement samples for each feature. Each tree casts a vote and the class that gets the maximum vote among the trees is used to categorize the samples [21].

C. Decision Tree

Decision tree is an easy, robust, and simple way of obtaining accurate decisions using a multistage approach. Multistage approaches deconstruct complex decisions into simpler decisions to obtain desired solutions more accurately and easily. A decision tree classifier is used on a dataset containing positive as well as negative labels and is insensitive to irrelevant features [22].

D. K-Nearest Neighbor (KNN)

To classify data from a given dataset, the KNN classifier employs the lesson names of the k most comparative neighbors to figure the course of the input archive. It does this by positioning the document's neighbors among the preparing records [23]. Equations (1) and (2) represent the formula of KNN.

$$d(x, y) = \sqrt{\sum_{i=1}^n (y_i - x_i)^2} \quad (1)$$

$$d(x, y) = (\sum_{i=1}^m |x_i - y_i|) \quad (2)$$

E. Logistic Regression

Logistic Regression is a statistical analysis method that makes use of prior observations from a dataset to forecast a binary result. By analyzing the link between one or more independent variables, the model forecasts the value of a dependent variable [24]. It is an easy-to-enforce method and effective to train. Logistic Regression is not recommended for compound datasets [6].

F. Multi-Layer Perceptron (MLP)

MLP classifiers are strong and adaptable tools for classification applications, but to get the optimum performance, tuning is required. MLP can create complex, non-linear relationships between features and target variables. It is versatile and can easily handle high-dimensional data well. Despite these features, MLP is prone to overfitting, and training for a large dataset using MLP can be computationally expensive.

G. XGBoost

Extreme Boosting algorithm or XGBoost algorithm, is a widely used machine learning algorithm that builds an ensemble of models with lower learning capacities and then applies the boosting strategy to get a powerful learner. It is particularly useful in situations involving several features and sample data points because of its capacity to manage large dimensionality and non-linearity in correlations [26]. Equations (3) give the mathematical formula of XGBoost.

$$y = \sum_{t=1}^m f(x) \quad (3)$$

H. AdaBoost

Adaptive boosting or AdaBoost is an algorithm used for improving the conduct of simple machine learning models in grouping tasks. It has a low generalization error and is easy to implement [25]. Figure 1 shows a schematic diagram of AdaBoost.

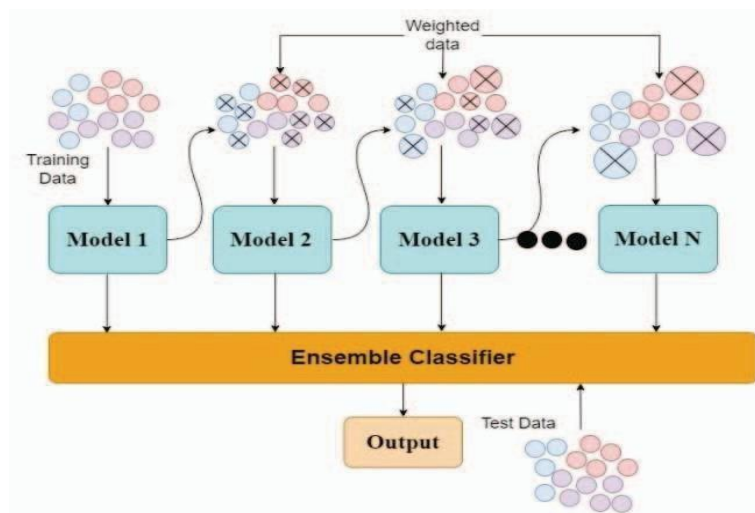


Fig.1. Schematic representation of AdaBoost

I. Gradient Boosting Machine (GBM)

Angle boosting machines, or GBMs, utilize a learning handle that fits modern models one after the other to deliver a reaction variable gauge that's more exact. This strategy is utilized to construct unused base-learners that have the most noteworthy conceivable relationship with the ensemble's negative slope of the misfortune work. Because of their great adaptability and their simplicity in implementation, the GBMs can be applied to almost any data-driven task [27].

J. Light Gradient Boosting Machine (LGBM)

Large datasets can be handled with Light GBM, a gradient-boosting framework that is quick and effective. In order to attain speed and accuracy, it makes use of feature parallelism and gradient-based one-sided sampling, as well as decision trees acting as weak learners.

K. Back Propagation Neural Network (BPNN)

A kind of multilayer artificial neural network trained with the backpropagation technique is called a Backpropagation Neural Network (BPNN). It reduces the error between the network's predictions and the intended outputs by iteratively adjusting the weights and biases of connected neurons.

5. EXPERIMENTATION AND RESULT

The implementation of the above 9 models is discussed in this section. The conduct of the models has been analyzed in a tabular form. The following section contains two tables that describe the performance of the ML models along with Confusion Matrix and ROC curves of the best fit models with and without PCA. The sub-sections are distributed as A. Implementation, B. Results and C Discussion.

A. Implementation

After completion of data exploration, the models were implemented on the preprocessed dataset. ROC curves and confusion matrix of all models were generated and the metrics in Table 1 were used for analyzing the performance of models. Table 1 also represents their acronyms.

TABLE I. METRICS USED

Sl no.	Metrics used	Abbreviation
1.	Accuracy	ACC
2.	Precision	
3.	Recall	
4.	F1 score	
5.	Confusion Matrix	CM
6.	True Positives	TP
7.	True Negatives	TN
8.	False Positives	FP
9.	False Negatives	FN

The formula for the above metrics is as follows. Equations (4), (5), (6), and (7) give the formulas for Accuracy, Precision, Recall, and F1 Score respectively.

$$ACC = \frac{TP+TN}{TP+FP+TN} \quad (4)$$

$$Precision = \frac{TP}{FP+TN} \quad (5)$$

$$Recall = \frac{TP}{TP+FN} \quad (6)$$

$$F1\ Score = \frac{2TP}{2TP+FP+FN} \quad (7)$$

B. Results

This section gives the final result of the experiment. It is divided into 2 sub-sections namely a) Analysis without PCA and b) Analysis with PCA.

a) Analysis without PCA

After analysing Table II, it was found that Arbitrary Woodland is the best-fit show with an exactness of 0.99. It outflanks all other models in exactness, review, and F1-score. It contains a exactness of 0.94 for false or '0' labelsand 1.0 for true or '1' labels. The recall is found to be 0.99and 0.98 for false and true values respectively while the F1- score is found to be 0.96 and 0.99 for false and true labelsrespectively.

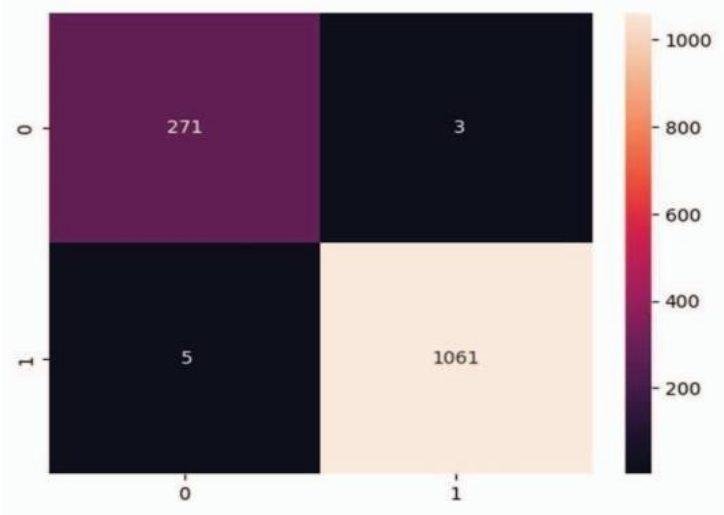


Fig. 2. Confusion Matrix of Random Forest without PCA

The ROC curve of the Random Forest shown in the figure is also found to be perfectly fitting. Figure 2 and Figure 3 show the Confusion Matrix and ROC curve of Random Forest.

Logistic Regression is found to be the worst-performing model among all with the lowest of 0.98. It has low Precision, recall, and F1 score values, especially for the false labels. Table II shows the metricsEvaluation Table of models not using PCA.

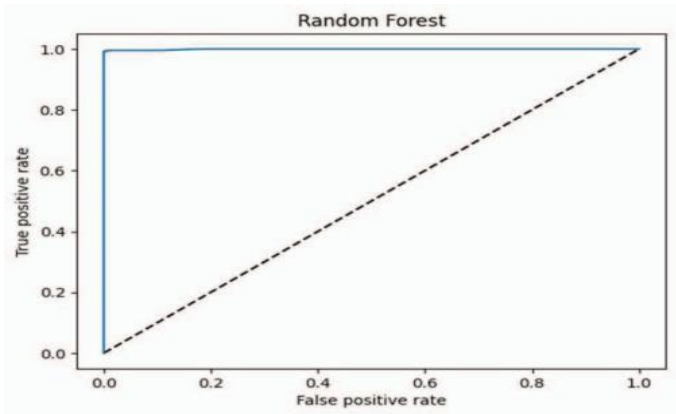


Fig.3. ROC curve of Random Forest without PCA

b) Analysis with PCA

On analyzing Table III, it was found that Random Forest is once again the best-fit model

but this timewith an exactness of 0.98. It outflanks all other models in precision exactness, review, and F1-score which are found to be 0.98 for both wrong and genuine names. Figure 4 and figure 5 speak to the Disarray Framework and the ROC bend of Arbitrary Timberland.

Logistic Regression is found to be the most noticeably awful- performing show among all with the least precision of 0.81. It has moo Exactness, review, and F1 score values. The Precision is found to be 0.80 and 0.82 for false and true labels. The recall values are found to be 0.83 and 0.79 for false and true labels respectively while the F1 score was obtained to be 0.81 for both false and true labels. Table III represents the Metrics Evaluation Table of models using PCA.

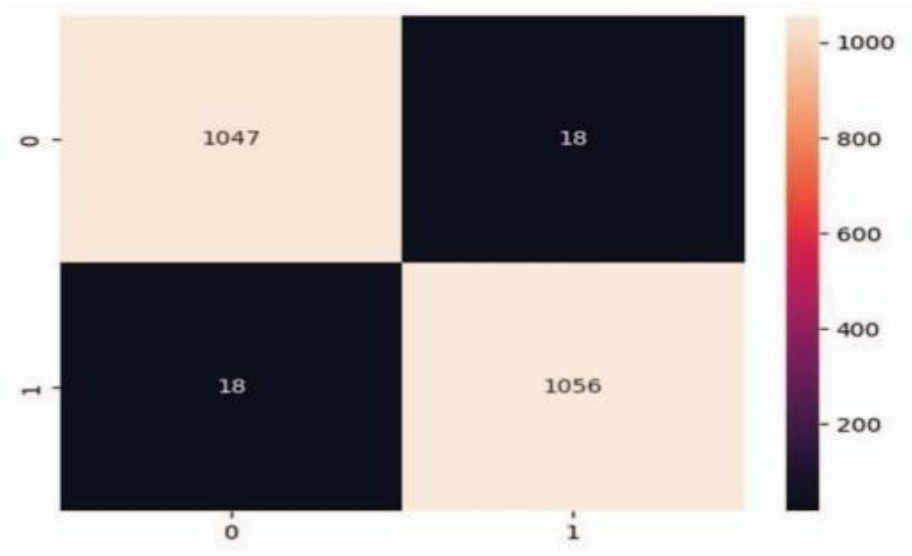


Fig.4. Confusion Matrix of Random Forest with PCA

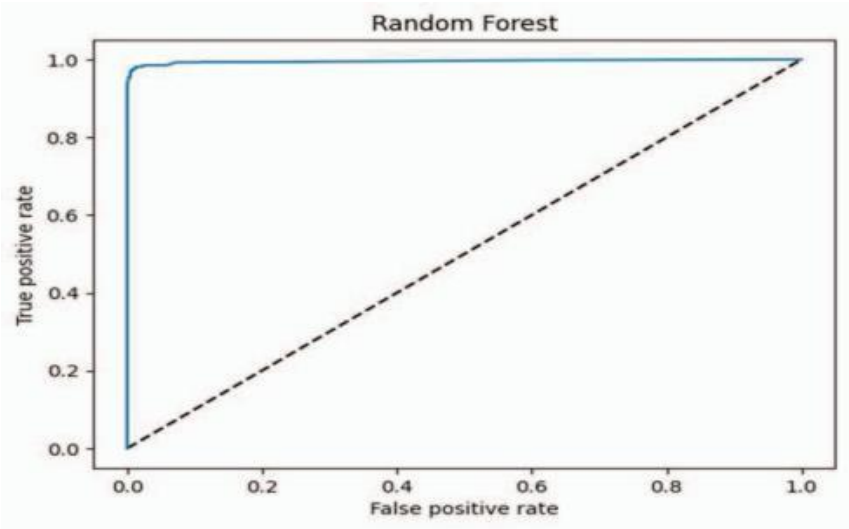


Fig.5. ROC curve of Random Forest with PCA

TABLE II. METRICS ANALYSIS TABLE WITHOUT PCA

Model	class	Accuracy	Precision	Recall	F1 Score	CM	
						TP	FP
						TN	FN
SVM	0	0.99	0.94	0.99	0.96	1050	4
	1		1.00	0.98	0.99	270	16
Random Forest	0	0.99	0.98	0.99	0.99	1061	3
	1		1.00	1.00	1.00	271	5
KNN	0	0.99	0.99	0.97	0.98	1062	8
	1		0.99	1.00	0.99	266	4
Logistic Regression	0	0.98	0.94	0.96	0.95	1050	11
	1		0.99	0.98	0.99	263	16
MLP	0	0.99	0.98	0.96	0.97	1061	11
	1		0.99	1.00	0.99	263	5
Decision tree	0	0.99	0.98	0.97	0.97	1061	9
	1		0.99	1.00	0.99	265	5
AdaBoost	0	0.98	0.97	0.95	0.96	1058	14
	1		0.99	0.99	0.99	260	8
Gradient Boosting	0	0.98	0.97	0.94	0.96	1058	16
	1		0.99	0.99	0.99	258	8
XGBoost	0	0.99	0.98	0.97	0.98	1061	7
	1		0.99	1.00	0.99	267	5
Light GBM	0	0.99	0.98	0.98	0.98	1061	5
	1		0.99	1.00	0.99	268	6
BPNN	0	0.99	0.98	0.95	0.97	1061	5
	1		0.99	1.00	0.99	261	13

C. Discussion

Random Forest is found to have a better performance than all other models with and without PCA. The high accuracy shows that the model is capable of predicting a higher percentage of true positives. This model is highly reliable for predicting results as the confusion matrix is perfectly balanced when using PCA. The high recall value and precision shows that the model barely misses any true positives. A high F1 score indicates that the model recognizes the genuine positives well and maintains a strategic distance from the wrong positives.

Moreover, the ensemble approach of Random Forest tends to reduce overfitting. The randomness of the model makes it less sensitive to specific training data which in turn helps it achieve a better performance on the testing data, thus outperforming all others.

Logistic Regression is found to have the worst performance than all other models with and without PCA. The use of PCA affected the performance of Logistic Regression, which had a significant drop in accuracy due to the reduction of features.

TABLE III. METRICS EVALUATION TABLE WITH PCA

Model	class	Accuracy	Precision	Recall	F1 Score	CM	
						TP	FP
						TN	FN
SVM	0	0.88	0.93	0.82	0.87	1008	66
	1		0.84	0.94	0.89	871	194
Random Forest	0	0.98	0.98	0.98	0.98	1056	18
	1		0.98	0.98	0.98	1047	18
KNN	0	0.98	0.99	0.96	0.98	1067	7
	1		0.96	0.99	0.98	1021	44
Logistic Regression	0	0.81	0.80	0.83	0.81	846	228
	1		0.82	0.79	0.81	885	180
	0		0.92	0.93	0.93	991	76

MLP	1	0.93	0.93	0.92	0.93	989	83
Decision tree	0	0.97	0.98	0.97	0.97	1054	35
	1		0.97	0.98	0.97	1030	20
AdaBoost	0	0.90	0.90	0.90	0.90	966	104
	1		0.90	0.90	0.90	961	108
Gradient Boosting	0	0.94	0.94	0.95	0.94	1005	52
	1		0.95	0.94	0.94	1013	69
XGBoost	0	0.98	0.98	0.98	0.98	1051	22
	1		0.98	0.98	0.98	1043	23
Light GBM	0	0.97	0.98	0.97	0.97	1049	25
	1		0.97	0.98	0.97	1035	30
BPNN	0	0.91	0.88	0.95	0.91	933	141
	1		0.94	0.87	0.90	1009	56

6. CONCLUSION AND FUTURE WORK

The Random Forest calculation outflanked all other models in terms of the ROC bend and attained an accuracy of 0.99 and 0.98 with and without PCA respectively. The K-Nearest Neighbor (KNN) model demonstrated an accuracy of 0.99 and 0.98 with and without PCA respectively. It falls in quick succession with the Random Forest classifier with the other parameters being less than Random Forest. Promising outcomes were also displayed by other models, such as Decision Tree, XG Boost, AdaBoost, MLP, SVM, and GBM. Logistic Regression was the worst-performing model among all. On applying PCA, its accuracy decreased further due to the decrease of dimensionality.

In future works, progressed strategies such as Normal Dialect Handling (NLP) and Profound Learning may be utilized to identify Malware. Normal Dialect Preparing can be utilized in programs requiring input from the user to detect and classify the Malware. The ML algorithms used in this paper are quite simple and their practical use on a real-world complex dataset may generate a lower accuracy and precision. However, in the future, Deep Learning Algorithms will be implemented on complex datasets to achieve more accurate AI-based detection.

REFERENCES

- [1] I. Santos, J. Devesa, F. Brezo, J. Nieves, and P. G. Bringas, "OPEM: A Static-Dynamic Approach for Machine-learning-based Malware Detection." International Joint Conference CISIS'12- ICEUTE' 12-SOCO' 12 special sessions, pp. 271-280, 2013.
- [2] N. Pachhala, S. Jothilakshmi, and B. P. Battula, "A Comprehensive Survey on Identification of Malware Types and Malware Classification Using Machine Learning Techniques," in 2021 2nd International Conference on Smart Electronics and Communication (ICOSEC), IEEE, pp. 1207–1214, 2021.
- [3] I. Firdausi, C. lim, A. Erwin, and A. S. Nugroho, "Analysis of Machine learning Techniques Used in Behavior-Based Malware Detection," in 2010 Second International Conference on Advances in Computing, Control, and Telecommunication Technologies, IEEE, pp. 201–203, 2010.
- [4] L. Liu, S. Zhao, H. Chen, and A. Wang, "A new machine learning method for identifying Alzheimer's disease," Simul Model Pract Theory, vol. 99, 2020.

- [5] M. K. Gourisaria, P. Jain, V. Singh, & T. Choudhury ., “A duplex method for classification of Parkinson's disease using data reduction techniques” in *Advances in Distributed Computing and Machine Learning: Proceedings of ICADCML*. Singapore: Springer Nature Singapore, 2022, pp. 555-565.
- [6] R. Pramanik, S. Khare, G.M. Harshvardhan, and M. K. Gourisaria, “A comparative study for depression prediction using machine learning classification models” in *Advances in Data and Information Sciences: Proceedings of ICDIS*. Singapore: Springer, 2022, pp. 233-246.
- [7] M. Poostchi, K. Silamut, R. J. Maude, S. Jaeger, and G. Thoma, “Image analysis and machine learning for detecting malaria,” *Translational Research*, vol. 194, pp. 36–55, 2018.
- [8] K. H. Miao, J. H. Miao, and G. J. Miao, “Diagnosing Coronary Heart Disease Using Ensemble Machine Learning,” *International Journal of Advanced Computer Science and Applications*. Vol. 7, No. 10, 2016.
- [9] U. Singh, M. K. Gourisaria, and B. K. Mishra, “A Dual Dataset approach for the diagnosis of Hepatitis C Virus using Machine Learning,” in *2022 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT)*, IEEE, pp. 1–6, 2022.
- [10] M. N. Ashtiani and B. Raahemi, “Intelligent Fraud Detection in Financial Statements Using Machine Learning and Data Mining: A Systematic Literature Review,” *IEEE Access*, vol. 10, pp. 72504– 72525, 2022.
- [11] S. K. Thangavel, P. D. Bkaratki, and A. Sankar, “Student placement analyzer: A recommendation system using machine learning,” in *2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS)*, IEEE, pp. 1– 5, 2017.
- [12] A. Kundu, P. Dutta, K. Ranjit, S. Bidyadhar, M. K. Gourisaria, & H. Das. “Software Fault Prediction Using Machine Learning Models”. In *2022 OITS International Conference on Information Technology (OCIT)*, pp. 170-175, 2022
- [13] H. Rathore, S. Agarwal, S. K. Sahay, and M. Sewak, “Malware Detection using Machine Learning and Deep Learning,” In *Big Data Analytics: 6th International Conference, BDA 2018, Warangal, India, Proceedings 6*, pp. 402-411, 2019.
- [14] J. Li, L. Sun, Q. Yan, Z. Li, W. Srisa-an, and H. Ye, “Significant Permission Identification for Machine-Learning-Based Android Malware Detection,” *IEEE Trans Industr Inform*, vol. 14, no. 7, pp. 3216–3225, 2018.
- [15] C.-Y. Huang, Y.-T. Tsai, and C.-H. Hsu, “Performance Evaluation on Permission-Based Detection for Android Malware,” In *Advances in Intelligent Systems and Applications-Volume 2: Proceedings of the International Computer Symposium ICS 2012*, Hualien, Taiwan, pp. 111–120, 2013.
- [16] R. S. Pircoveanu, S. S. Hansen, T. M. T. Larsen, M. Stevanovic, J. M. Pedersen, and A. Czech, “Analysis of Malware behavior: Type classification using machine learning,” in *2015 International Conference on Cyber Situational Awareness, Data Analytics and Assessment (CyberSA)*, IEEE, pp. 1–7, 2015.

- [17] S. Saad, W. Briguglio, and H. Elmiligi, "The Curious Case of Machine Learning In Malware Detection," in 5th International Conference on Information Systems Security and Privacy, May 2019.
- [18] R. Sharma, M. K. Gourisaria, and S. S. Patra, "Cloud Computing— Security, Issues, and Solutions," In Communication Software and Networks: Proceedings of INDIA, pp. 687–700, 2021.
- [19] Parthajit Borah; DK Bhattacharyya; JK Kalita, 2021, TUNADROMD, 25th Dec, 2023, <https://www.kaggle.com/datasets/joebeachcapital/tunadromd-malware-detection>
- [20] S. V. M. Vishwanathan and M. N. Murty, "SSVM: A simple SVM algorithm" in Proc. 2002 International Joint Conference on Neural Networks. IJCNN'02, vol. 3. IEEE, May, 2002.
- [21] M. Pal, "Random forest classifier for remote sensing classification," Int J Remote Sens, vol. 26, no. 1, pp. 217–222, 2005.
- [22] S. Chandra, M. K. Gourisaria, H. Gm, S. Rautaray, M. Pandey, and S. N. Mohanty, "Semantic Analysis of Sentiments through Web- Mined Twitter Corpus." In ISIC (pp. 122-135), 2021.
- [23] S. TAN, "An effective refinement strategy for KNN text classifier," Expert Syst Appl, vol. 30, no. 2, pp. 290–298, 2006.
- [24] T. G. Nick and K. M. Campbell, "Logistic regression" in Topics in Biostatistics, 2007, pp. 273-301.
- [25] L. Pallippattu Mathai, "Malware detection on android using Adaboost algorithm MSc Research Project Msc Cyber Security." (Doctoral dissertation, Dublin, National College of Ireland), 2021.
- [26] S. Ghosh, M. K. Gourisaria, B. Sahoo, and H. Das, "A pragmatic ensemble learning approach for rainfall prediction," Discover Internet of Things, vol. 3, no. 1, p. 13, 2023.
- [27] A. Natekin and A. Knoll, "Gradient boosting machines, a tutorial," Front Neurorobot, vol. 7, 2013