*Short communication*

Enhanced Malware Detection in windows Application Using Ensemble

Learning Technique

.

ABSTRACT

Detecting malware is essential for defending computer systems against attacks that jeopardize their operation and security. Detecting malware is essential for defending computer systems against attacks that jeopardize their operation and security finding and eliminating harmful software that jeopardizes computer systems' availability, confidentiality, or integrity is the aim of malware detection. In the constantly changing field of cybersecurity, identifying malware in applications. Accuracy and efficiency are crucial. The goal of this research is to improve malware detection capabilities in applications by utilizing machine learning methods. Adaboost, a contemporary machine learning (ml) technique, will be used in this work to investigate malware detection and classification components. The "windows malware detecion dataset," a publicly accessible dataset, is used in the proposed study. To choose the optimal features, we used an ensemble learning approach. Our goal is to create a reliable detection system that incorporates cutting-edge machine learning methods. This project uses machine learning algorithms, such as supervised and unsupervised learning approaches, to create learning methods. The proposed model

Uses machine learning algorithms, such as supervised and unsupervised learning approaches, to create a model that can categorize and recognize. To create a reliable detection system, proposed methodology consists of data preparation, model training, and evaluation. The proposed methodology outperforms existing models with an accuracy of 99.9%.

KEYWORDS— *ADABOOST, MALWARE, AND DECISION TREES.*

INTRODUCTION

Malware poses a danger to computer systems' stability and security, necessitating more sophisticated detection strategies than only conventional signature-based approaches. By identifying malicious software through the analysis of patterns and anomalies, machine learning (ml) provides a potent remedy. The windows malware detection dataset is used in this study to improve malware classification accuracy with adaboost, an ensemble learning technique. For better detection, we combine static and behavioural analysis through feature selection, data preparation, and model training. Our objective is to create a reliable system that effectively classifies malware by utilizing supervised learning, advancing cybersecurity and fortifying defences against changing cyber threats.

Conventional malware detection systems mostly use heuristic and signature-based techniques to categorize files as either benign or harmful. These methods, however, have trouble with significant false positives and novel malware types. By combining adaboost with decision trees, our suggested solution improves detection and allows for multi-class classification of malware families. It uses ensemble learning to increase accuracy while extracting characteristics from several sources. While other ml algorithms are highlighted in related work, our method provides a high-accuracy, scalable solution. We give cybersecurity professionals comprehensive insights for focused mitigation tactics by classifying different types of malware.

2. MATERIAL AND METHODS

the proposed model is a scalable, yet powerful deep learning architecture model based on adaboost as depicted in fig-1 to classify different families of malware with high efficiency. by utilizing adaboost in conjunction with decision trees for multi-class classification, proposed model enhances malware

detection. proposed methodology divides malware into different families, in contrast to conventional systems that just detect the existence of malware.

* 1. *data preprocessing*: the preprocessing stage involves extracting critical attributes from malware samples, including pe\_header, pe\_section, dlls\_imported, and api\_functions. this step ensures data consistency by handling missing values, encoding categorical variables, and normalizing or scaling features. to enhance model generalization and performance, the dataset is systematically split into training, validation, and test sets.
	2. *model training*: in the training phase, decision trees serve as weak learners, while adaboost enhancesinstances. decision trees develop classification rules based on extracted features, whereas adaboost refines performance iteratively. key hyperparameters, such as the number of estimators and tree depth, are optimized to achieve the best results.
	3. model evaluation: to assess the classifier’s effectiveness, the trained model is tested using unseen data. performance metrics such as accuracy, precision, recall, and f1-score are computed. additionally, new malware samples are introduced to evaluate the model’s robustness, simulating real-world conditions to ensure reliable detection across various malware families.
	4. implementation: the model relates to cybersecurity systems, used for real-time malware detection, and retrained on a regular basis to accommodate changing threats.

to guarantee resilience, the model is tested with fresh samples after being trained on an extensive dataset. this method offers a high-accuracy, scalable solution that helps cybersecurity experts detect and eliminate different malware threats.

Picture 1: Malware detection dataset



3. RESULTS AND DISCUSSION

the analysis of the proposed model was performed on the windows malware detection dataset containing windows portable executable (pe) samples, categorized into four distinct feature sets. each feature set is stored in a separate csv file, providing detailed insights into various aspects of malware behaviour.

dlls\_imported.csv: lists the dynamic link libraries (dlls) imported by each malware sample. the first column contains sha256 hash values, the second represents the malware family label, and the remaining columns detail the imported dll names.

api\_functions.csv: records the api functions invoked by the malware, including their corresponding sha256 hashes and labels.

pe\_header.csv: contains 52 labelled fields that describe the pe header attributes of the executable files.

pe\_section.csv: features values from 9 fields across 10 different pe sections, all appropriately labelled.

the dataset consists of six malware family labels: 0 – benign, 1 – redlinestealer, 2 – downloader, 3 – rat, 4 - banking trojan, 5 – snakekeylogger, 6 – spyware.

 

figure-2 accuracy

fig-2 in the accuracy graph denotes that the proposed model denotes that the accuracy of k-nearest neighbours (knn), decision tree (dt), random forest (rf), neural network, and adaboost are 89.22%, 99.70, 98.47%, 32.52, and 100% respectively. the results highlight the effectiveness of ensemble methods in improving malware detection accuracy. outperforms existing models. this suggest that it is capable of learning and generalizing of complex feature patterns.



figure-3 precision

the precision demonstrated in fig-3 measures the precision of positive malware classes in the model. the precision scores for different models are k-nearest neighbours (knn) - 88.57%, decision tree (dt) - 99.78%, random forest (rf) - 98.53%, neural network - 72.7%, and adaboost - 100%. the findings demonstrate that ensemble techniques significantly enhance detection precision.

 

figure-4 recall

the recall depicted in the fig-4 demonstrates that all the related malware samples are correctly identified by the models. the recall scores for different models are as follows: k-nearest neighbours (knn) - 87.81%, decision tree (dt) - 99.67%, random forest (rf) - 98.27%, neural network - 28.79%, and adaboost - 100%. the results highlight the effectiveness of ensemble techniques in improving recall performance for malware detection.



figure-5 f1-score

the f1-score depicted in the fig-5 takes both, precision and recall measures to determine the system classification effectiveness. the f1 score is 1.00 of adaboost proves that the model has equally good performance in all the metrics and can thus be considered the most accurate of the models for malware detection and classification.

5.conclusion

this study highlights the importance of malware detection in safeguarding computer systems from security threats. By leveraging machine learning techniques, particularly adaboost, the research enhances the accuracy and efficiency of malware classification. using the windows malware detection dataset, an ensemble learning approach was applied to optimize feature selection and improve detection performance. the proposed model, integrating both supervised and unsupervised learning methods, effectively classifies and identifies malware. the results demonstrate that the developed system surpasses existing models, achieving an impressive accuracy of 99.9%, making it a highly effective solution for malware detection.

COMPETING INTERESTS DISCLAIMER:

Authors have declared that they have no known competing financial interests OR non-financial interests OR personal relationships that could have appeared to influence the work reported in this paper.

REFERENCES

garg, n. tahilramani, p. chauhan and r. patel, "microsoft malware detection using machine learning," 2022 6th international conference on intelligent computing and control systems (iciccs), madurai, india, 2022, pp. 1-6, doi: 10.1109/iciccs53718.2022.9788420.

akhtar, muhammad & feng, tao. (2022). malware analysis and detection using machine learning algorithms. symmetry. 14. 2304. 10.3390/sym14112304.

usha divakarla, k hemant kumar reddy, k chandrasekaran, a novel approach towards windows malware detection system using deep neural networks, procedia computer science, volume 215, 2022, pages 148-157, issn 1877-0509, <https://doi.org/10.1016/j.procs.2022.12.017>.

maray, mohammed, mashael s. maashi, haya mesfer alshahrani, sumayh s. aljameel, sitelbanat abdelbagi and ahmed s. salama. “intelligent pattern recognition using equilibrium optimizer with deep learning model for android malware detection.” ieee access 12 (2024): 24516-24524.

ayeni, olaniyi & owolafe, otasowie & akinsola, olabiyi. (2021). malware detection using machine learning. 84-87. 10.20533/icitst.2021.0008.

ogwara no, petrova k, yang ml, macdonell sg. mindpres: a hybrid prototype system for comprehensive data protection in the user layer of the mobile cloud. sensors. 2025; 25(3):670. <https://doi.org/10.3390/s25030670>.

guo w, du w, yang x, xue j, wang y, han w, hu j. malhapgnn: an enhanced call graph-based malware detection framework using hierarchical attention pooling graph neural network. sensors. 2025; 25(2):374. <https://doi.org/10.3390/s25020374>.

anderson, hyrum s., and phil roth. "ember: an open dataset for training static pe malware machine learning models." arxiv preprint arxiv:1804.04637 (2018).

alazab, mamoun & venkatraman, sitalakshmi & watters, paul. (2010). towards understanding malware behaviour by the

extraction of api calls. proceedings - 2nd cybercrime and trustworthy computing workshop, ctc 2010. 10.1109/ctc.2010.8.

alazab, m., venkatraman, s., watters, p., alazab, m., alazab, a. (2012). cybercrime: the case of obfuscated malware. in: georgiadis, c.k., jahankhani, h., pimenidis, e., bashroush, r., al- nemrat, a. (eds) global security, safety and sustainability & e- democracy. e-democracy icgs3 2011 2011. lecture notes of the institute for computer sciences, social informatics and telecommunications engineering, vol 99. springer, berlin, heidelberg. <https://doi.org/10.1007/978-3-642-33448-1_28>.

w. huang and j. w. stokes, "mtnet: a multi-task neural network for dynamic malware classification", proc. int. conf. detection intrusions malware vulnerability assessment, pp. 399-418, jul. 2016.

raff, e., barker, j., sylvester, j., brandon, r., catanzaro, b., & nicholas, c. (2017). malware detection by eating a whole exe. arxiv preprint arxiv:1710.09435.

krcál, marek, ondř švec, martin bálek and otakar jasek. “deep convolutional malware classifiers can learn from raw executables and labels only.” international conference on learning representations (2018).

pachhala, n., jothilakshmi, s., & battula, b. p. (2023). enhanced malware family classification via image-based analysis utilizing a balance-augmented vgg16 model. traitement du signal, 40(5), 2169-2178.

pachhala, n., jothilakshmi, s., battula, b.p. (2024). cross- platform malware classification: fusion of cnn and gru models. international journal of safety and security engineering, vol. 14, no. 2, pp. 477-486. https://doi.org/10.18280/ijsse.140215

1