Malware Detection using Machine Learning

Keya Dobriyal
[1]Amity University, Noida, INDIA

Author (e-mail: keya@ keyadobriyal.in).

This comprehensive analysis constitutes the final project submitted for the bachelor's degree in computer science (Honors in Data Science) at Amity University, Noida, India.

ABSTRACT

The rapid proliferation of interconnected computing nodes, including millions of Internet of Things (IoT) devices, has made cyber-attacks and evolving malware the most urgent threat to global information security. Among these threats, malware remains one of the most pervasive and damaging, exploiting system vulnerabilities to steal, alter, or destroy data. Traditional signature-based detection methods, while effective against known threats, often fail to address the challenges posed by rapidly evolving malware variants and sophisticated obfuscation techniques such as code insertion, instruction substitution, and data encoding. This limitation has highlighted the need for more adaptive and intelligent detection mechanisms.

This study explores the efficacy of this integration, utilizing an open-source dataset to train and evaluate various ML algorithms for identifying complex malware signatures. By leveraging the ability of ML models to learn from data, identify hidden patterns, and adapt to new threats, researchers and practitioners can significantly enhance cybersecurity defenses. The study provides an overview of common machine learning approaches for malware detection, compares their performance, and highlights the potential of ML-driven systems to strengthen future malware defense strategies.

By providing a foundational overview of ML algorithms suitable for this domain and presenting performance analysis, this research demonstrates that ML-driven anti-malware systems offer a necessary, continuous learning, and highly effective enhancement to modern cybersecurity measures.

INDEX TERMS

Malware detection, Machine learning, Cybersecurity, Classification algorithms, Obfuscation techniques, Supervised learning, Feature extraction, Intrusion detection systems

I. INTRODUCTION

Cyber-attacks are currently the most urgent issue in the realm of information sharing and data transfer over the internet. The Internet has become integral to our daily lives, serving as a major channel for exchanging information between numerous computing nodes, which include millions of interconnected computers and Internet of Things (IoT) devices. Consequently, this extensive information network has become a prime target for cybercriminals, with malware being one of the most significant threats to cyberspace.

The term "cyberattack" refers to exploiting a system's vulnerability for malicious purposes, such as theft, alteration, or destruction. **"Malware" combines 'mal' from malicious and 'ware' from software, representing programs or instructions designed to exploit users, businesses, or institutions.** Malware encompasses a variety of threats, including viruses, trojan horses, ransomware, spyware, adware, rogue ware, vipers, scareware, and more (Kaspersky Lab). By definition, malware is a piece of code that infiltrates a system and executes itself without the user's knowledge or consent. Over the last decade, there has been an 87% increase in malware infections and potentially unwanted programs.

Recently, the risk of financial fraud using malware has surged, compelling individuals in finance, organizations, and security companies to employ automated or semi-automated online methods for malware detection and removal. These systems analyze fraud trends to develop efficient detection mechanisms. Malware detection focuses on scanning systems and files to identify malicious software, leveraging advanced tools and techniques for effectiveness.

The primary aim of this term paper is to explore the application of machine learning techniques for detecting and analyzing malware. Traditional signature-based detection methods, though somewhat effective, struggle to keep up with the rapidly evolving malware variants and their sophisticated obfuscation techniques, such as Exclusive-OR, base 64 encoding, ROT 13, dead code insertion, instruction substitution, and sub-routing reordering (Shaukat et al., 2020). These techniques introduce malicious elements into binary and textual data, making it challenging for some malware detectors to interpret and identify (Shaukat et al., 2022).

In response to these challenges, integrating machine learning techniques in malware analysis and detection has emerged as a promising approach to enhancing cybersecurity measures. Cybersecurity experts have demonstrated that ML-driven anti-malware software is highly effective at detecting and neutralizing malware, as these systems continuously learn and improve to evade anti-malware defenses.

II. BREIF LITERATURE REVIEW

Cyberattacks orchestrated by hackers are currently the foremost concern in today's interconnected world. Over the past decade, cybercrime has burgeoned into a $1.5 trillion industry, operating with the sophistication of legitimate businesses.

Systematic malware analysis has proven crucial in identifying patterns indicative of malicious intent and preempting future threats. This analysis aims to examine distinct features and potential impacts of such threats, whether in the form of software or code.

The primary goal of malware analysis is to extract pertinent patterns from suspicious files to effectively combat cybersecurity threats. Following this analytical process, the subsequent step involves developing machine learning models tailored for malware detection, leveraging identified patterns or features that distinguish malware from benign programs. The ultimate outcome is a malware detection technique capable of identifying and neutralizing programs or files intended for malicious activities.

Malware analysis typically encompasses four stages: Static Property Analytics, Interactive Behavior Analytics, Manual Code Reversing, and Fully Automated Analysis. Various types such as static, dynamic, and hybrid malware analysis may be employed individually or in combination.

In the early days of cybersecurity, manual filtering rules sufficed due to low incident rates of malware attacks. Today, however, cybercrime poses the primary global business risk, with data breaches averaging a cost of $4.5 million, and cloud computing implicated in nearly 82% of cases (Artic Wolf). In 2024 alone, 35.9 billion records were breached across 9,478 publicly disclosed incidents (IT Governance, May 2024), underscoring the necessity for advanced security technologies.

Machine learning, particularly deep learning, holds promise in revolutionizing malware detection within the field of artificial intelligence. By analyzing extensive datasets, machine learning systems can infer patterns and make accurate predictions or decisions. Training algorithms on large datasets comprising both clean and malicious files enables them to adapt to the evolving nature of malware.
This term paper draws on relevant published literature to comprehensively explore and address these topics.

III. BASICS OF MACHINE LEARNING

Machine learning

Machine learning represents a paradigm shift by utilizing computational algorithms to identify patterns and anomalies within large datasets, thereby enabling the automated detection and classification of malware samples. As defined by AI pioneer Arthur Samuel, machine learning encompasses methods that provide computers with "the ability to learn without being explicitly programmed." In essence, a machine learning algorithm detects, discovers, and formulates underlying principles from the data it processes and learns from.

A machine learning model is a mathematically formalized set of principles capable of identifying patterns or making decisions based on previously unseen data. Machine learning techniques employ various approaches to uncover pattern-based solutions rather than relying on a single prediction method. Consequently, in the ever-evolving landscape of data exploitation, these techniques are highly adaptable, performing diverse tasks and improving from the data they encounter.

Classification of machine learning techniques

Machine learning is widely classified into four major categories, which are based on the nature of the learning process and the type of problems that they aim to solve.

Supervised Machine Learning Algorithms

Supervised Learning is a machine learning algorithm that uses labelled datasets as input to train a model. The main objective of this algorithm is to learn the relationship between input data and output labels, enabling it to make predictions or classifications on new, unseen data. The following methods are included in supervised learning:

a. Regression
b. Classification

Unsupervised Machine Learning Algorithms

Unsupervised learning algorithms analyze data to identify patterns without human intervention or predefined instructions. The machine autonomously detects correlations and relationships by examining the training dataset. The algorithm aims to organize unlabelled data and establish a systematic structure for it. The methods that fall under unsupervised learning include:

a. Clustering
b. Dimensionality Reduction

Semi supervised Machine Learning Algorithms

Semi-supervised learning combines elements of both supervised and unsupervised learning by utilizing both labelled and unlabelled data. Labelled data contains meaningful tags that guide the learning process, and the algorithm attempts to learn from this data to label the unlabelled data. It includes the following methods:

a. Text classification

Reinforced Machine Learning Algorithms

Reinforcement learning emphasizes structured learning processes, continually updating its knowledge within a framework of predefined rules, actions, parameters, and objectives. Within these rules, the algorithm explores various options and possibilities, monitoring and evaluating each outcome to identify the optimal one. This approach teaches the machine to use trial and error to achieve the best possible result.

Teaching the machine to learn

Standard machine learning follows a cyclical process consisting of four main stages. It begins with data management, where a training dataset is gathered. After collecting the data, it undergoes preprocessing and exploration to understand its structure and significance. The data is then divided into training and validation subsets.

The next stage involves training machine learning models, selecting appropriate algorithms for the task. Following this, the third stage assesses model outputs using various methods and algorithms. Evaluation metrics such as F1 score, precision, recall, and accuracy rate are used, with the efficiency of algorithms measured using a confusion matrix to determine the best approach.
Moving to the final stage, the model is applied to new data, and outcomes are monitored. Continuous learning and improvement occur throughout this process, ensuring refinement until the cycle restarts.

Data input and Validation

The initial step in every machine learning process is the Data Input phase. During this stage, data is structured, cleaned, and prepared to proceed to subsequent steps. Data validation involves assessing statistical distributions and measurements, such as range, number of categories, subgroup distributions, and others.

Pre-processing of data

Data pre-processing is a crucial phase in machine learning. This procedure involves preparing raw data for use by the machine learning model. It includes evaluating, filtering, removing inaccuracies, and filling in

missing data to eliminate issues and ensure the data is suitable for machine learning purposes. The processed dataset is then divided into two subsets: one for training the model and another for validating its performance

Data model training and Validation

In this phase, the model is trained to predict outputs accurately based on inputs (pre-processed datasets) using fitting algorithms. Various machine learning algorithms fall into three main categories: Supervised, Unsupervised, and Reinforcement learning, which are employed for constructing data models. Each model is evaluated using metrics like F1 score, precision, recall, and accuracy rate. The efficiency of the algorithms is assessed using a confusion matrix to determine the optimal method.

Deployment of model

The ultimate stage of a machine learning model involves deploying it after training and analysis for practical application in real-world scenarios. The primary objective in machine learning is integrating models into operational environments. Model deployment can be achieved through three methods: a model server, a web browser, or an edge device.

IV. SELECTION OF MACHINE LEARNING

Choosing the appropriate machine learning model involves a blend of art and science, influenced by factors such as the size, quality, and diversity of data, as well as the specific analytical objectives derived from that data. It becomes essential to employ an experimental and iterative approach to identify the most effective method in terms of performance, accuracy, reliability, and actionable insights. Each type of model possesses distinct strengths and weaknesses in learning and prediction, prompting the combination of multiple machine learning. types and various algorithms within those types to achieve optimal outcomes.

Over the past two decades, researchers have predominantly utilized one or a combination of four main types of machine learning models, selected based on their suitability for data preparation methods (George Lawton):

a. Supervised learning models: Utilize pre-labelled data provided by humans.
b. Unsupervised learning models: Discover patterns in data without prior labelling.
c. Semi-supervised learning models: Employ an iterative process capable of handling both labelled and unlabelled data.
d. Reinforcement learning models: Use algorithms to make decisions through trial-and-error learning processes to achieve optimal results.

In this study, I employed several machine learning classifiers, specifically K-Nearest Neighbor (KNN), Decision Tree, Random Forest, and an Artificial Neural Network (ANN), to maximize detection accuracy.

V. PROBLEM STATEMENT

- Ongoing cyberattacks by hackers represent a significant challenge in the realm of data analytics and transfer in today's technology-driven world.
- Traditional antivirus systems that rely on signature matching often fail to detect polymorphic and highly obfuscated executables.
- Static analysis based on human heuristic inspection has become unreliable due to the rapid evolution of malware.
- The study focuses on detecting malware in a downloaded dataset containing files, utilizing various machine learning algorithms.

- Models are evaluated and compared based on metrics such as accuracy, efficiency, and F-1 Score.
- Ultimately, the best-performing model is determined after thorough comparison.
- Ethical considerations and challenges associated with the use of machine learning for malware detection and analysis are addressed.

VI. DATA SET

The Malimg dataset serves as a standard benchmark comprising grayscale images representing various malware families. It contains 9,939 samples across 25 different malware families: each derived from the binary structure of the respective malware files. These images vary in size, ranging from 64 pixels by 200 pixels to 800 pixels by 800 pixels.

For experimental purposes, a subset of the dataset was selected, featuring 10,868 instances categorized into 9 classes. Each binary file was interpreted as an array of 8-bit unsigned integers and organized into a two-dimensional array format. Training utilized 80% of the total data, while the remaining 20% was allocated for cross-validation.

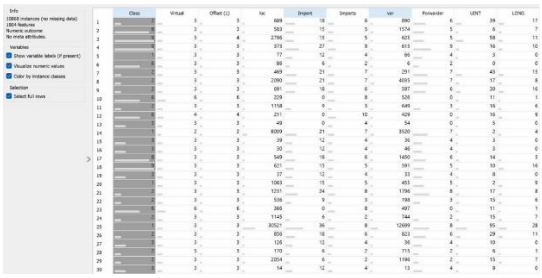

Fig 1. Data set attributes used for machine learning

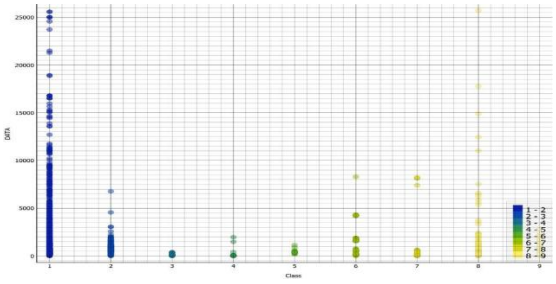

Fig 2. Data distribution based on class of malware

VII. METHODOLOGY

This term paper introduces four machine learning algorithms used in the workflow for detecting and classifying malware. Traditional malware detection historically relies on two main techniques: static detection and dynamic detection, commonly employed by antivirus companies.

Static malware analysis, also known as signature-based detection, is a proactive security approach that examines software without executing it. It focuses on scrutinizing the structure, behavior, and content of files to identify potential malicious code.

Dynamic malware detection involves running suspicious programs in a controlled environment, such as a virtual machine or sandbox, to observe their real-time behavior.

The preprocessed dataset was then inputted into four distinct models created using the open-source machine learning and data visualization software, ORANGE. These models include K-Nearest Neighbors (KNN), Neural Network, Decision Tree, and Random Forest. They were selected because they are widely used in machine learning and provide a solid benchmark for comparison.

These machine learning techniques were applied to the dataset to predict the presence of malware.

KNN [Supervised Learning]

The K-Nearest Neighbors (KNN) algorithm is a non-parametric supervised learning classifier that utilizes proximity measurements to classify or predict the grouping of a given data point. Initially developed by Evelyn Fix and Joseph Hodges in 1951, it was further developed by Thomas Cover. KNN finds applications primarily in pattern recognition, data mining, and intrusion detection.

Artificial Neural Network [Reinforcement Learning]

An Artificial Neural Network (ANN) is a computational model inspired by the structure of the human brain, specifically its neurons or nodes. It comprises a multitude of nodes organized into layers, including input and output layers at opposite

ends, and multiple hidden layers in between. These interconnected nodes collaborate to solve specific problems.

ANNs are sophisticated, nonlinear statistical models designed to uncover intricate patterns. They learn from examples and experience, making them adept at handling high-dimensional data with complex relationships among input variables. Additionally, ANNs can be trained using sample data rather than the entire dataset, making them particularly valuable for modeling non-linear relationships.

Decision Tree [Supervised Learning – Classification]

A decision tree is a tree-like structure resembling a flowchart. In this structure, each internal node denotes a feature or attribute, branches represent decision rules based on those attributes, and each leaf node signifies a potential outcome of a decision. Every node in the tree corresponds to a test on a specific variable, and each branch signifies the result of that test. The decision tree is a non-parametric supervised learning model that predicts the target variable's value by learning straightforward rules derived from data attributes through classification and regression techniques.

Random Forest [Supervised Learning – Classification]

Random forests, also known as "random decision forests," utilize ensemble learning, a concept that combines multiple decision trees trained on different subsets of the dataset to achieve improved results in classification and regression tasks. Each individual tree in the forest may be weak on its own, but their combination enhances predictive accuracy significantly. The algorithm begins with individual decision trees; predictions from each tree are aggregated to produce the final output. Similar to a forest, a greater number of trees results in higher accuracy due to the diverse perspectives and collective decision-making process of the ensemble.

VIII. METHODOLOGY

The subset of dataset consisted of nine categories of malware and set of non-malware files, the use of machine learning techniques for training utilizing classifier discussed in methodology above. The entire simulation is carried out on Mac Air using open-source software “Orange”. The results of these tests sorted on accuracy achieved during testing phase, along with an analysis is presented herewith.

K-**Nearest Neighbor**: An accuracy of 95.9% was achieved during training phase and 96.3 % was achieved during testing phase.

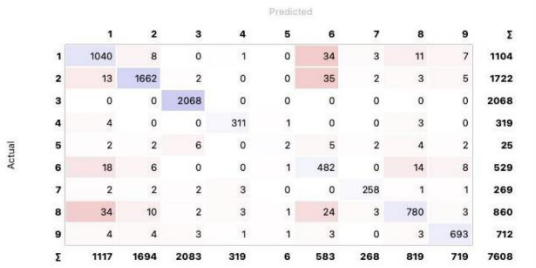

Fig 3. Confusion matrix during training phase using KNN ML algorithm

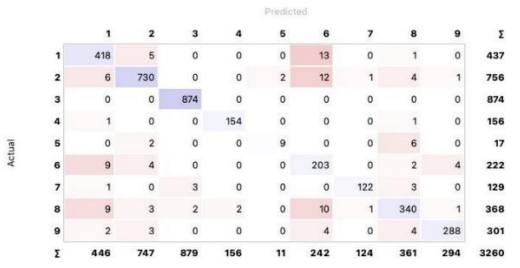

Fig 4. Confusion matrix in testing phase using KNN ML algorithm

Decision Tree: An accuracy of 97.7% was achieved during training phase as well as testing phase

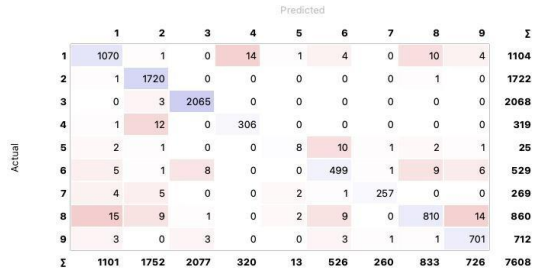

Fig 5. Confusion matrix during training phase using DT ML algorithm

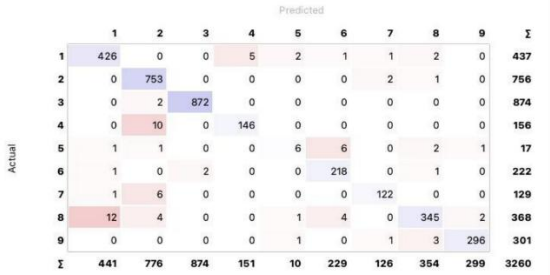

Fig 6. Confusion matrix in testing phase using DT ML algorithm

Neural Network: An accuracy of 98.8% was achieved during training phase and 99.1 % was achieved during testing phase.

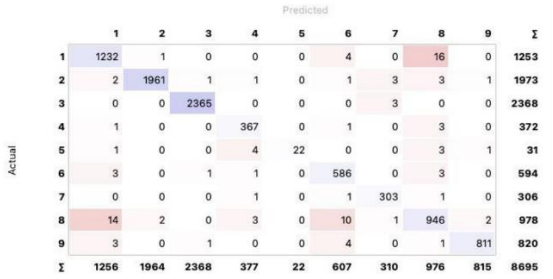

Fig 7. Confusion matrix during training phase using ANN ML algorithm

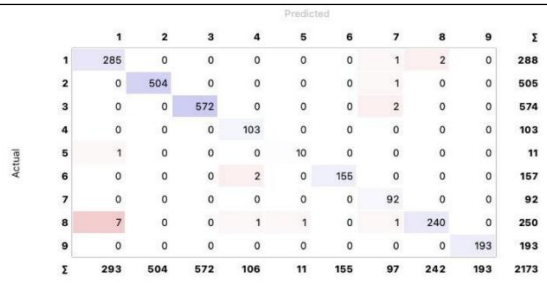

Fig 8. Confusion matrix in testing phase using ANN ML algorithm

Random Forest: An accuracy of 99.1% was achieved during training phase and 99.4 % was achieved during testing phase.

Fig 9. Confusion matrix during training phase using RF ML algorithm

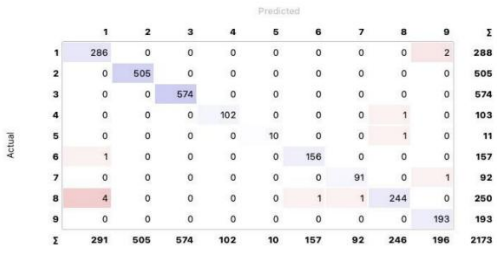

Fig 10. Confusion matrix in testing phase using RF ML algorithm

As random forest algorithm was found to be the most accurate method, the following ROC figures are of the training phase.

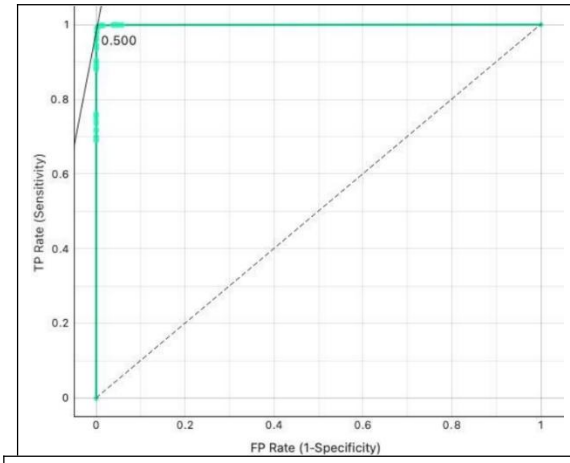

Fig 11. ROC 1 of RF algorithm

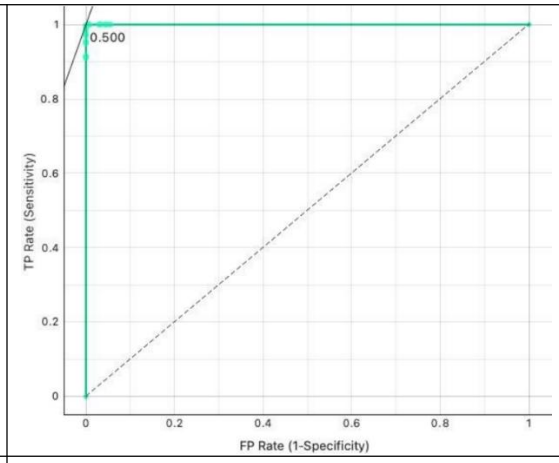

Fig 12. ROC 2 of RF algorithm

IX. Conclusion and Results

Machine learning has revolutionized cybersecurity by offering adaptable and dependable techniques to detect and eliminate malicious software. Systems employing supervised and unsupervised learning algorithms can now detect previously unknown threats, adapt to evolving malware behaviors, and provide real-time protection. Techniques such as classification, clustering, and anomaly detection enable the creation of models capable of analyzing vast datasets, identifying patterns, and accurately predicting potential threats. Integrating these technologies promises to enhance cybersecurity defenses' effectiveness and responsiveness, despite challenges like the need for large, high-quality datasets and the complexity of interpreting machine learning models.

In the term paper, a dataset was utilized to train four machine learning models using supervised and reinforcement learning techniques. The K-Nearest Neighbor, Decision Tree, Neural Network, and Random Forest achieved accuracies of 96.3%, 97.7%, 99.1%, and 99.4%, respectively. Notably, the Random Forest model emerged as the most accurate, achieving 99.1% accuracy in training and 99.4% on test data.

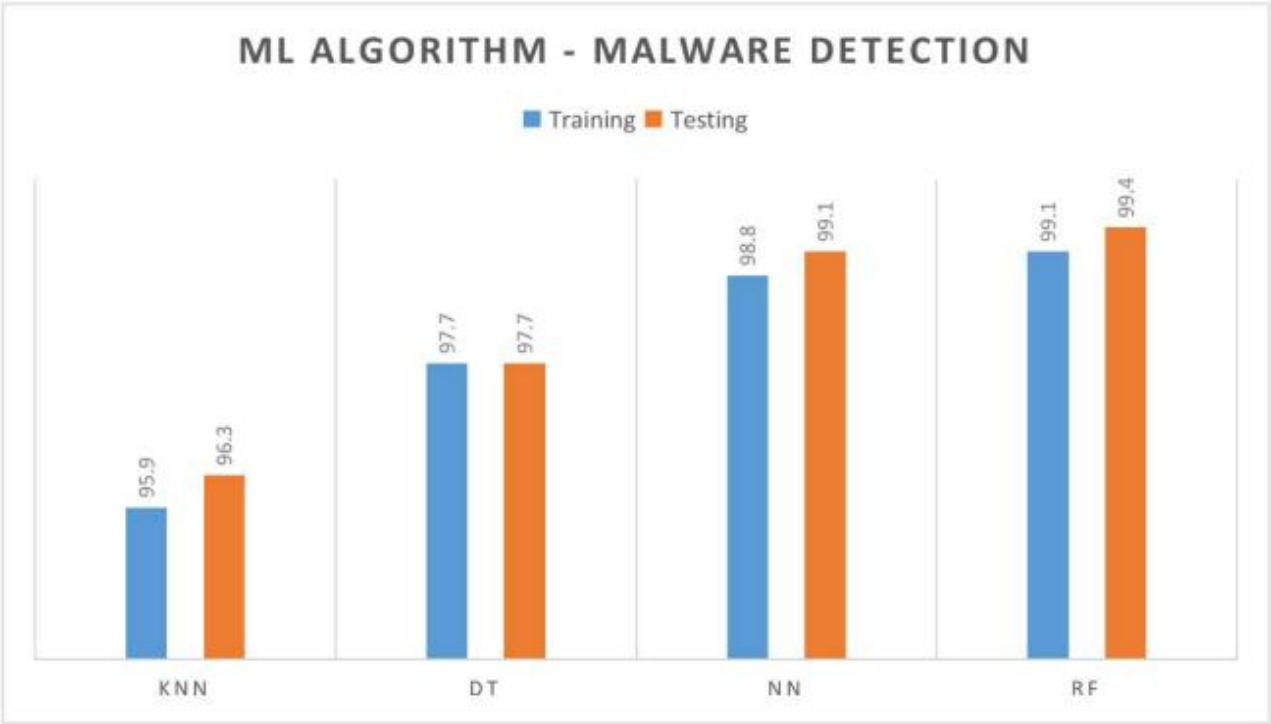

Fig 20. Comparison of accuracy of malware detection

Machine learning algorithm	Accuracy achieved in training phase	Accuracy achieved in testing phase
K-Nearest neighbour	95.9	96.3
Decision tree	97.7	97.7
Neural Network	98.8	99.1
Random forest	99.1	99.4

The result obtained in the experiment has achieved an accuracy of 99%, which was obtained using random forest approach of classification. Confusion matrix analysis shows that other three methods were not efficient to detect class 8 malware than the random forest which could handle detection of all classes of malware.

X. References

[1] Akshit Kamboj, Priyanshu Kumar, Amit Kumar Bairwa, Sandeep Joshi: Detection of malware in downloaded files using various machine learning models: Egyptian Informatics Journal 24 (2023) 81-94

[2] Pascal Maniriho, Abdun Naser Mahmood, Mohammad Jabed Morshed Chowdhury: A systematic literature review on Windows malware detection: Techniques, research issues, and future directions: The Journal of Systems and Software 209 (2024) 111921

[3] Austin Brown, Maanak Gupta, Mahmoud Abdelsalam: Automated machine learning for deep learning-based malware detection: Computers and Security 137 (2024) 103582

[4] Damien Warren Fernando, Nikos Komninos: FeSAD ransomware detection framework with machine learning using adaption to concept drift: Computers & Security 137 (2024) 103629

[5]S. Poornima a, R. Mahalakshmi: Automated malware detection using machine learning and deep learning approaches for android applications: Measurement:

Sensors 32 (2024) 100955

[6]Huijuan Wang, Boyan Cui, Quanbo Yuan, Ruonan Shi, Mengying Huang: A review of deep learning-based malware detection techniques: Neurocomputing 598 (2024) 128010

[7]Kamran Shaukat, Suhuai Luo, Vijay Varadharajan: A novel deep learning-based approach for malware detection: Engineering Application of Artificial Intelligence 122 (2023) 106030

[8]Kamran Shaukat, Suhuai Luo, Vijay Varadharajan: A novel machine learning approach for detecting first-time-appeared malware: Engineering Applications of Artificial Engineering 131 (2024) 107801

[9]Daniel Gibert, Jordi Planes, Carles Mateu, Quan Le: Fusing feature engineering and deep learning: A case study for malware classification: Expert Systems with Applications 207 (2022) 117957

[10]Daniel Gibert, Carles Mateu, Jordi Planes: The rise of machine learning for detection and classification of malware: Research developments, trends and challenges: Journal of Network and Computer Applications 153 (2020) 102526

[11]Ihab Shhadat, Bara' Bataineh, Amena Hayajneh, Ziad A. Al-Sharif: The use of machine learning technique to advance the detection and classification of unknown malware: International Workshop on data driven Security 2020, Warsaw, Poland

[12]Sudesh kumar, Shersingh, Siddhant kumar, Karan Verma: Malware Classification Using Machine Learning Models: International Conference on Machine Learning and Data Engineering (ICMLDE 2023)

[13]Muhammad Azeem, Danish Khan, Saman Iftikhar, Shaikhan Bawazeer, Mohammed Alzahrani: Analyzing and comparing the effectiveness of malware detection: A study of machine learning approaches: Heliyon 10 (2024) e23574

[14]Espoir K. Kabanga, Chang Hoon Kim: Malware images classification using Convolutional Neural Network: Journal of Computer and Communications, 2018, 6, 153-158

[15]Muhammad Shoaib Akhtar and Tao Feng: Malware Analysis and Detection Cornell UniversityUsing Machine Learning Algorithms: Symmetry, Nov 2022

[16]Ahmed Bensaoud, Nawaf Abudawaood, and Jugal Kalita: Classifying Malware Images with Convolutional Neural Network Models: Cornell University, arxiv:2010.16108

About Author:

Keya Dobriyal is currently pursuing the B.E. degree in Computer Science with Honors in Data Science from Amity University Noida. She has completed few projects in data science, machine learning, deep learning. Data-driven problem solver with a passion for turning raw data into valuable business intelligence, with hands-on academic experience in data mining, predictive modeling, and data visualization. Skilled in leveraging Python and R to extract actionable insights from complex datasets.