Virus Detection Software

Neha Barde Ritisha Kumar Akshata Ubhale

Usha Mittal Institute of Technology
SNDT Women's University, Mumbai.

Mini Project Assessment Phase 1

October 1, 2024

Outline of Topics

1. Introduction to Project: Problem Statement
2. Objectives of this Project
3. Literature Survey
4. Discussing approach and methodology
5. Hardware and Software Requirements
6. References

Introduction

Cyber attacks are a major concern in modern technology, targeting system vulnerabilities to steal, alter, or destroy data. These threats are increasingly sophisticated, making it essential to use advanced methods for detecting and preventing them. Thus the need to involve Machine learning (ML), which offers a promising solution for enhancing real-time malware detection.
PROBLEM STATEMENT: *Improve virus detection accuracy and efficiency in large-scale datasets using a Random Forest-based machine learning model.*

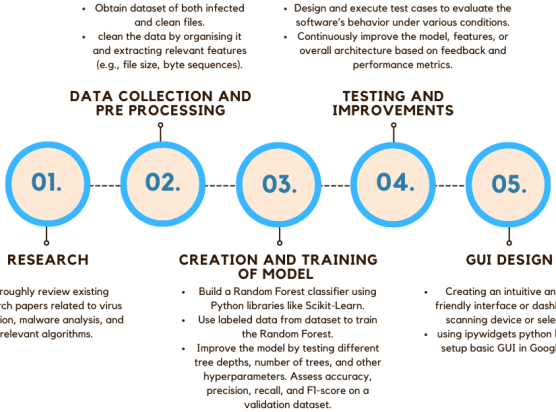

Malware
Man In the Middle
Phishing
Drive-By Downloads
Password Attacks
Malvertising
DDoS
Rogue Software

Objectives of the Project

In this project, we aim to build a virus detection system that addresses key limitations found in studies of our Literature Survey. We are currently focusing on enhancing accuracy better than existing systems, which is essential for reliable virus detection in diverse environments.

1. Improve dataset diversity to enhance model generalization across various scenarios, by taking a large dataset.
2. Reduce computational cost to enable more complex models without compromising efficiency.
3. Ensure applicability to real-world data for reliable performance outside of controlled or simulated environments, by taking datasets from various real life sources which include legitimate and virus and infected files.

Architecture



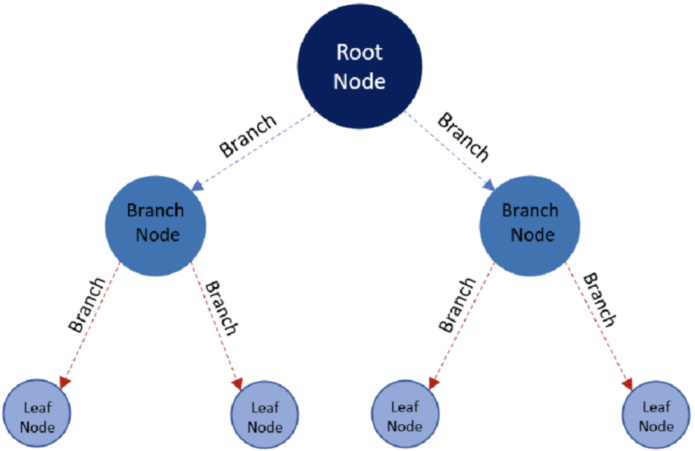
Literature Survey

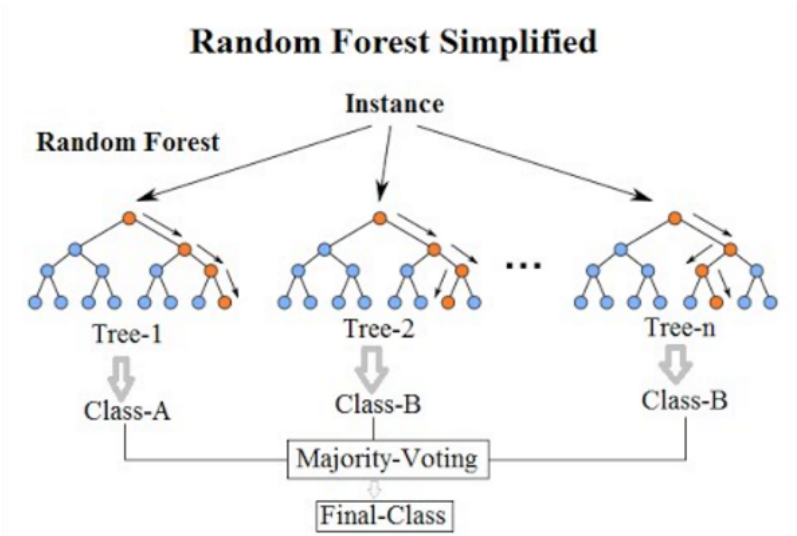
TITLE	AUTHORS	YEAR	KEY FINDINGS	LIMITATIONS
Malware detection using machine learning	Dragoș Gavriluț et al.	2009	Early exploration of using machine learning for identifying malware in systems.	Limited by the computational power available at the time, affecting the complexity of models that could be implemented.
Malware Detection Using Machine Learning: A Review	M. S. Hossain et al.	2016	Reviews various ML methods and their applications in malware detection.	Lacks practical implementation examples, making it more theoretical than applicable.
An Empirical Evaluation of Machine Learning Algorithms for Malware Classification	M. C. Tan et al.	2017	Compares accuracy and performance of several ML algorithms, including Random Forest.	Limited dataset diversity, potentially affecting the generalizability of the results.
A Comprehensive Review of Random Forest Applications	Jones and Liu	2020	Random Forest models improve malware detection accuracy by up to 25%	Focuses only on Random Forest, without considering other ensemble methods that might offer improved

Study on Malware Detection Using ML	Smith et al.	2021	Traditional methods miss up to 30% of new malware variants. ML models show improved detection rates.	The study does not consider the computational cost of implementing complex ML models in real-time systems.
Detection of Malware in Downloaded Files Using Various Machine Learning Models	Akshit Kamboj et al.	2022	Highlights the effectiveness of different machine learning models in identifying malware.	Limited by the scope of the dataset used, which may not represent all types of malware encountered in real-world scenarios.
Enhancing Malware Detection with Machine Learning	Brown et al.	2022	ML models, including Random Forest, achieve accuracy rates over 90%.	The paper does not address the interpretability of the models, which is critical for cybersecurity applications.
Enhancing Malware Detection Through Machine Learning Techniques	Zeina S. Jassim et al.	2024	Discusses the improvement of malware detection accuracy through advanced ML techniques.	The study's reliance on simulated data may limit its applicability in real-world malware detection scenarios.

Random Forest Classifier

Random Forest is a powerful machine learning algorithm used for classification and regression tasks. It works by building multiple decision trees during training and combining their results to improve the model's overall prediction accuracy and reliability. Each tree is trained on a random subset of data, and the final prediction is typically made through majority voting with respect to classification.


Root Node
Branch
Branch
Branch Node
Branch Node
Branch
Branch
Branch
Branch
Leaf Node
Leaf Node
Leaf Node
Leaf Node


Random Forest Simplified
Instance
Random Forest
Tree-1
Tree-2
Tree-n
Class-A
Class-B
Class-B
Majority-Voting
Final-Class

Hardware and Software Requirements

Hardware Requirements:

- Processor: Intel i3 or better
- Memory(RAM): 8GB RAM
- Storage: 256GB or up hardware, we have used upto 3GB maximum for database storage and program.

Software Requirements:

- Operating system: Windows 10 or up, MacOS
- Programming Language: Python 3.x
- Machine Learning Libraries: Pandas, NumPy, Sci-kit Learn
- IDE: Jupyter Notebook

Research Papers and Books referenced:

1. D. Gavriluț et. al. "Malware detection using machine learning", International Multiconference on Computer Science and Information Technology, 2009, , doi: 10.1109/IMCSIT.2009.5352759.
2. Akshit Kamboj et. al. "Detection of malware in downloaded files using various machine learning models" Egyptian Informatics Journal, 2023, https://doi.org/10.1016/j.eij.2022.12.002.
3. Zeina S.Jassim et. al. "Enhancing Malware Detection Through Machine Learning Techniques", InfoTechSpectrum: Iraqi Journal of Data Science (IJDS), 2024, https://doi.org/10.51173/ijds.v1i1.4

Book Mastering Machine Learning for Penetration Testing by Chiheb Chebbi