International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
Article Received:*25 July 2023* ***Revised:****12 September 2023* ***Accepted:****30 September 2023*

Malware Detection Techniques based on Machine Learning

Dhruv Singh Rajput
Department of Computer Science and Engineering
SSET, Sharda University
Greater NOIDA, U.P. India
dhruvsinghrajput2002@gmail.com

Gouri Sankar Mishra
Department of Computer Science and Engineering
SSET, Sharda University
Greater NOIDA, U.P. India
gourisankar.mishra@sharda.ac.in

Ayush Pratap Singh
Department of Computer Science and Engineering
SSET, Sharda University
Greater NOIDA, U.P. India
ayushpsingh14@gmail.com

Pradeep Kumar Mishra
Department of Computer Science and Application
SSET, Sharda University
Greater NOIDA, U.P. India
pradeepkumar.mishra@sharda.ac.in

Abstract— Artificial intelligence and machine learning have become crucial tools in the fight against cyber attacks. With the constant evolution of technology, traditional methods of protecting networks are no longer enough. This is where AI and machine learning come into play, by analyzing vast amounts of data and detecting patterns or anomalies that might indicate a potential threat. This paper aims at understanding and analyzing the implementation of Artificial Intelligence (AI) and Machine Learning (ML) systems in enhancing cyber security. By detecting patterns and anomalies in network traffic, AI algorithms can quickly identify potential threats and reduce response time, far surpassing human capabilities. This not only saves valuable time and resources for organizations but also improves overall protection against cyber-attacks. As technology continues to advance, it is crucial that we leverage AI for cybersecurity to stay ahead in the fight against malicious actors. With proper utilization of AI and ML technologies, we can ensure a safer digital future for all users..

Keywords- Malware; anti-malware; machine learning; feature extraction; feature selection; random forest; SVM; neural networks

I. INTRODUCTION

Malware, malicious software designed to harm users, poses a growing cyber threat in today's internet landscape. While traditional signature-based antivirus struggles to detect new and unseen threats, the ease of acquiring malware development tools and pre-made software lowers the barrier for potential attackers. Malware itself employs sophisticated techniques like polymorphism and automatic updates to evade detection. Machine learning offers a promising alternative, analyzing patterns to identify previously unseen malware, but faces challenges with certain types like polymorphic malware and requires high-quality training data. This highlights the need for continuous research and development in detection methods, security practices, and user awareness to combat the evolving threat of malware.

A serious cyberthreat known as malware has grown to be a commonplace on the internet, a ground-breaking instrument that has revolutionized communication and information access. These harmful applications, which can range from espionage tools to information-stealing programs, can cause serious harm to unwary users. Malware is software that is expressly created to penetrate user computers and inflict harm in many ways, as defined appropriately by Kaspersky Labs.

Due to its reliance on signature-based identification, traditional anti-virus software faces a serious threat from the fast development of malware. Using this method, files are compared to a sizable database of recognized virus signatures. Unfortunately, this method's efficacy is severely limited because it can only recognize malware that has already been encountered. Considering the startling rate at which new malware variants appear—estimates indicate that hundreds of thousands are produced every day—this presents a sizable blind area.

The declining barrier to expertise in malware generation exacerbates the limits of signature-based detection. Because dangerous tools are easily accessible online and pre-made malware can be purchased on the black market, even those with no technical knowledge might potentially become attackers. Studies like as Aliyev (2010) demonstrate this tendency, which suggests a rise in automated assaults as well as attacks carried out by less technically skilled persons, who are sometimes referred to as "script-kiddies."

In addition, malware is increasingly utilizing advanced evasion strategies to elude conventional detection methods. Many malware programs now use methods like automated upgrades to keep ahead of current signatures and polymorphism, in which the virus continuously modifies its signature to avoid detection. The efficacy of signature-based detection is decreasing due to these developments.

Machine learning has emerged as a viable method for malware detection in order to get around these restrictions. Machine learning does not depend on pre-defined signatures, in contrast to signature-based techniques. Instead, it is trained on large datasets of files that are both benign and dangerous, which enables it to recognize patterns and traits that mark harmful programs apart from those that aren't. This gives machine learning a major advantage over conventional techniques by allowing it to identify malware that has never been seen before.

However, it is crucial to acknowledge that even machine learning is not a silver bullet. While it can effectively detect many types of malware, it still faces challenges in identifying

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
Article Received:*25 July 2023* ***Revised:****12 September 2023* ***Accepted:****30 September 2023*

certain categories like polymorphic malware that constantly change their signatures. Additionally, the accuracy of machine learning models can be impacted by factors like the quality of training data and the ever-evolving nature of malware itself, leading to potential false positives (mistakenly identifying benign files as malicious) and false negatives (failing to detect actual malware).

The vital necessity for ongoing study and improvement in this subject is highlighted by the limitations of current detection technologies and the rapid evolution of malware. Our defenses must keep up with the increasing sophistication of cyber threats by utilizing a range of cutting-edge detection techniques, strong security protocols, and user awareness to successfully counteract the persistent threat posed by malware.

II. LITERATURE REVIEW AND RESEARCH ANALYSIS

It is now necessary to investigate other strategies in the battle against cyber threats due to the fast development of malware and the shortcomings of conventional signature-based detection techniques. Examining the literature on malware detection techniques, this study concentrates on the drawbacks of signature-based detection and the intriguing possibilities of machine learning (ML) as a strong substitute.

A. The Inevitable Shortcomings of Signature-Based Detection

Traditional anti-virus software's main feature, signature-based detection, uses pre-defined signatures to identify harmful software. Although this method works well against known viruses, it has a number of serious drawbacks:

- Lack of capacity to detect unknown malware: Research by Sebastián et al. (2016) has shown that signature-based techniques are unable to identify novel and unknown malware variants, leaving users open to new threats. Given that malware is constantly changing—estimates indicate that hundreds of thousands of new versions appear every day—this is a significant disadvantage.
- High false positive/negative rates: Research on the difficulties of preserving accuracy while using signature-based techniques is covered in studies such as Baskaran & Ralescu (2016). These techniques have the potential to produce false positives, which would classify innocuous files as dangerous by mistake, causing needless difficulties and interruptions. They also run the danger of producing false negative results, which leave systems open to possible assaults by failing to identify genuine malware.

These restrictions highlight the intrinsic flaws in signature-based detection, making it more and more ineffectual in the face of a dynamic threat scenario.

B. Machine Learning: A Promising Paradigm Shift

Machine learning has become a viable substitute method for malware detection in recent times. ML algorithms are more advantageous than signature-based techniques in a number of ways because they can learn from and adapt to data:

- Expertise in pattern recognition: McGraw & O'Hallaron's (2012) study shows that machine learning models are highly proficient at recognizing patterns inside data. This gives them a major edge over depending only on pre-defined signatures as it enables them to examine other aspects of a file, such code structure, behavior, and network activity, to identify malware that has never been seen before.
- Adaptability: ML models have the ability to continually learn and adjust to new threats, in contrast to signature-based techniques. ML models may be retrained on fresh datasets in order to keep up with developing threats, even when new malware types appear.

But it's important to recognize that machine learning isn't a panacea. The paper correctly highlights the continued difficulties that machine learning faces, especially with:

- Malware that is polymorphic: To avoid detection, this kind of malware modifies its code continuously, which makes machine learning models difficult to use. The continued need for improvements in managing polymorphic and previously undiscovered variations is highlighted by research by Wang et al. (2020).
- Excellent training data: The caliber and variety of the data that machine learning models are trained on have a significant impact on the models' efficacy. Research conducted by Biggio et al. (2012) highlights the need of utilizing superior training data to guarantee the precision and applicability of machine learning models.

The flexibility and capacity of machine learning to learn from data, in spite of these drawbacks, gives it a major edge over signature-based techniques. With further study and development, machine learning (ML) has enormous potential to transform malware detection and improve our defenses against online threats.

C. A Glimpse into the Future: Research Directions and Beyond

The final section of the article emphasizes the need of ongoing study and advancement in this important area:

- ML algorithm evolution: Researchers are always looking for new approaches to solve the drawbacks of the models they currently use. Technological developments in domains like as deep learning and anomaly detection have potential for managing obscure and multivariate malware (Wang et al., 2020).
- Improving training data: Techniques for obtaining high-quality, varied, and current training data are essential for enhancing machine learning models' efficacy and generalizability (Biggio et al., 2012). Collaboration between academics, experts in the industry, and security researchers may be necessary for this.
- A layered defense: According to McHugh et al. (2017), combining ML-based detection with other security measures like sandboxing and user education can provide a more complete protection against malware. We may build a more effective and durable defense against the ever-present danger of cyberattacks by utilizing a multi-layered strategy.

We can improve our defenses against malware and make sure that everyone uses the internet safely by tackling these issues. An overview of the drawbacks of signature-based detection and the promise of machine learning as a viable substitute strategy has been given by this study of the literature. Machine learning is set to become increasingly important in the ongoing fight against cyber threats as research in this area develops.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
Article Received:*25 July 2023* ***Revised:****12 September 2023* ***Accepted:****30 September 2023*

III. EASE OF USE

The two categories of malware detection strategies are behavior-based and signature-based. It is crucial to comprehend the fundamentals of both static and dynamic malware analysis techniques before delving into these techniques. Static analysis is carried out "statically," that is, without executing the file, as the name suggests. On the other hand, dynamic analysis is carried out on the file while it is being run, like in the sandbox or virtual machine.

A. Signature Based Malware detection

Traditional anti-virus software's main feature, signature-based detection, finds malware by comparing files to a database of known harmful code snippets, also referred to as signatures or hashes. This is how it operates:

- Scanning: The anti-virus program's scanner reads a file into your computer and pulls out a unique identifier, usually a hash, from the file's code.
- Comparison: Next, the extracted hash is compared by the scanner to a locally stored or cloud-accessible signature database. Millions of known malware signatures are present in this database.
- The verdict: The file is marked as malicious by the scanner if it discovers a match in the database. After that, the anti-virus program takes appropriate action, such erasing, blocking, or quarantining the file.
- Database Update: Security researchers examine newly found malware to extract its distinct signature. The anti-virus program can now recognize the new danger in the future as this signature has been added to the database.

Strengths of Signature-Based Detection:

- Simple and efficient: This solution is reasonably easy to implement and takes few system resources.
- Effective in thwarting recognized threats: Detection methods based on signatures are effective against known and reported malware variations.
- Low false positives: This technique seldom misclassifies innocent files as malware since it produces very few false positives when using a well-maintained database.

Weaknesses of Signature-Based Detection:

- Inefficient against hidden threats: This method ignores novel and hidden malware variations. The threat environment is continuously changing, and signature-based detection is unable to keep up with the rapid innovation of hackers producing new malware.
- Vulnerable to strategies of evasion: Certain malware uses strategies like packing (encryption) and polymorphism (repeated code modifications) to evade signature-based detection.
- Big database size: Updating a large database with millions of signatures on a regular basis might be resource-intensive.

In summary, even while signature-based detection is still an essential part of malware security, its shortcomings call for research into complementary and alternative strategies. With its capacity for adaptation and learning, machine learning is showing promise as a means of identifying new and hidden dangers.

B. Behavior-Based Malware Detection: Delving Deeper

Although it provides a first line of security, signature-based detection is unable to keep up with the constantly changing nature of malware. Behavior-based detection becomes an essential supplementary strategy as hackers create increasingly complex and novel threats. This approach, in contrast to concentrating on certain signatures, examines a program's real behavior in order to detect harmful intent.

a) Monitoring: A number of methods are used by behavior-based detection to keep an eye on program activities, such as:

- System calls: Tracking the calls an application makes to the operating system can help identify questionable behavior, such as efforts to change system settings or access private files.
- Network activity: Unauthorized access attempts, data exfiltration, and contact with known hostile servers can all be found by analyzing network traffic.
- File system access: Tracking efforts to open, edit, or remove files can reveal attempts to steal confidential information or interfere with important system components.

b) Analysis: Next, a predetermined set of behavioral patterns linked to malicious software are compared to the observed activity. These tendencies might involve efforts to:

- Introduce code to different processes.
- Turn off any security software.
- Inappropriately create or alter system configuration files.
- Interact with command-and-control servers that are recognized.

c) Verdict: The program's unusual activity, suggestive of malware, is detected by the system based on the study. If so, a notice is sent out and possible measures like deletion, quarantine, or halting execution may be taken.

Advantages of Behavior-based Detection:

- Detects unknown threats: By concentrating on behavioral patterns rather than predetermined signatures, this technique is able to detect novel and undetected malware types.
- Adapts to changing threats: The detection patterns change along with the behavior of the malware, enabling constant adjustment to the shifting threat environment.
- Effective against methods of evading signatures: Because behavior-based detection concentrates on the activities of the program rather than its precise code, it is less vulnerable to evasion strategies like polymorphism.

Challenges of Behavior-based Detection:

- False positives: It can be difficult to reliably discern between benign and malevolent activity, which may result in false positives and the banning of genuine applications.
- High in resources: Real-time program activity analysis and monitoring might demand a lot of system resources, which can affect performance.
- Complexity: To sustain efficacy, behavior-based detection must be implemented and adjusted with skill and ongoing observation.

Despite its difficulties, behavior-based detection is essential in the battle against contemporary malware. It provides a strong

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
***Article Received:**25 July 2023 **Revised:**12 September 2023 **Accepted:**30 September 2023*

tool for detecting and thwarting even invisible threats by examining program activity rather than depending just on signatures. To build a complete defense against cyberattacks, behavior-based detection must be used in conjunction with other security measures like user education and signature-based detection.

IV. Implementation Approach

This project aims to identify malicious software and URLs using machine learning. The focus is twofold:

- PE Malware Detection: In order to identify malicious Portable Executable (PE) files which are often utilized by Windows programs a Random Forest algorithm must be trained.
- URL Classification: In this case, the prediction of whether a particular URL is dangerous or not is done using a linear regression technique.

Both methods rely on labeled examples from datasets—both benign and malicious—to train the models, which are then used to forecast unknown data. The ultimate objective is to develop models for increased security that can precisely identify possible threats.

Essentially, machine learning tasks entail teaching algorithms to carry out particular tasks, such regression or, as in this instance, categorization. The given dataset serves as the basis for this training, and the final model may be used to additional data to provide predictions. The model's precise results will vary depending on the job selected and carried out.

To develop a deeper understanding, it is worth going through the general workflow of the machine learning process.

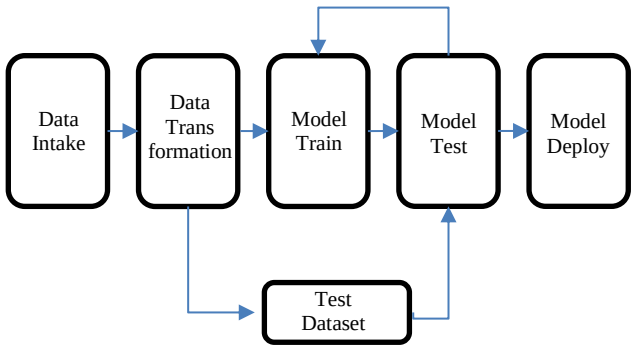

Figure 1. Steps followed in methodology

A. *Feature Extraction*

a) Portable Executable File Format

On Windows systems (32- and 64-bit), the Portable Executable (PE) file format is a commonly used data structure for a variety of file formats, including:

- Executable files (.exe)
- Object code (.obj)
- Dynamic Link Libraries (DLLs, .dll)
- Font files (.fon)
- Core dumps

PE functions essentially as a container format, holding the data required for the Windows OS loader to handle the code that is wrapped within the file. In Windows NT computers, this format took the place of the previous COFF (Common Object File Format).

b) Structure of a PE File:

A PE file can be broadly divided into two main sections:

- Header: This part includes important details about the file, including its size, the point at which execution starts, and any dependencies it has on other libraries.
- Sections: This part is further divided into more manageable subsections, each including code, resources (pictures, icons), and data (variables), among other specialized data components.

Understanding the PE file format is valuable for various purposes, including:

- Reverse engineering software
- Malware analysis
- Creating custom Windows applications

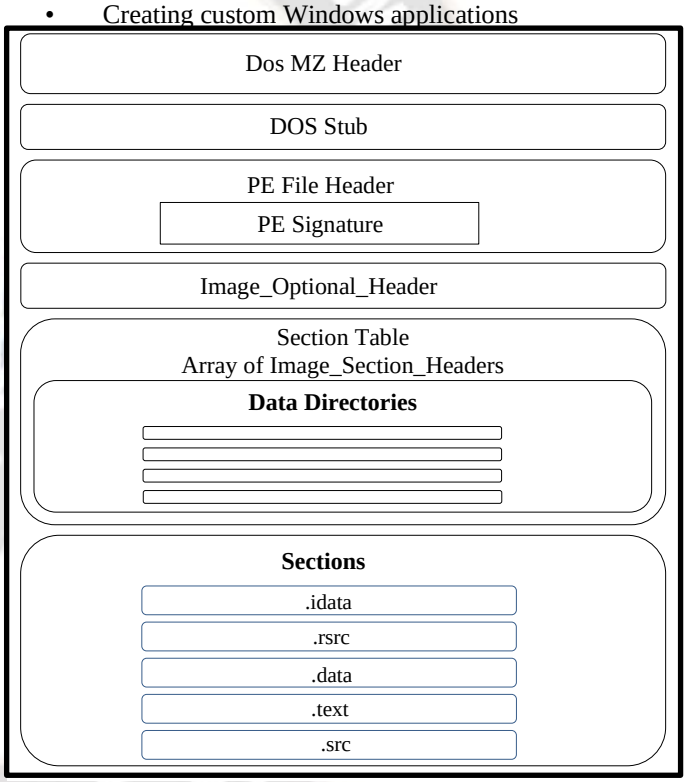

Figure 2. Structure of PE file

Despite its seeming simplicity, the header of the PE file format contains important hints for possible virus detection. Although the first MS-DOS stub helps with format recognition, the crucial e_lfanew value directs us to the file's genuine PE header.

This header contains key insights:

- Signature: Identifies the file as a PE format.
- Machines: Indicates the target architecture (e.g., x86, x64). Incompatibility might raise suspicion.
- NumberOfSections: An abnormally high number compared to typical applications could be a red flag.
- SizeOfOptionalHeader: An unusually large size might warrant further investigation.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
Article Received:*25 July 2023* ***Revised:****12 September 2023* ***Accepted:****30 September 2023*

These characteristics can help identify fraudulent PE files when used with machine learning algorithms. For effective malware security, a thorough strategy integrating a variety of detection techniques is still essential.

c) Malicious URL

Think of a URL as a unique address that helps you find anything on the internet, like a specific webpage, image, or video. It's made up of different parts, like:

- Protocol: Protocols, which are similar to languages, instruct browsers on how to communicate with websites (e.g., "http" for regular websites, "https" for secure websites).
- Domain Name: This is the name of the website, such as wikipedia.org or google.com.
- Path: See it as a website's internal navigation that points you to particular files or pages.

But not all URLs are created equal. Some bad guys use malicious URLs to trick you into clicking on them. These can lead to:

- Downloading malicious software, such as ransomware, malware, or viruses that take control of your computer.
- False webpages: Although these appear authentic, their purpose is to obtain your private data, such as credit card numbers or passwords (phishing).

So, how do you stay safe? Here are some tips:

- Treat connections you don't know well with suspicion: Nothing that appears strange or appears to be from someone you don't know should be clicked.
- Prior to clicking, make sure: Before you click on the link, move your cursor over it to view the exact URL.
- Continue using safe websites: Check the URL for "https" at the beginning, especially if you are submitting personal data.
- Maintain software updates: This assists in patching vulnerabilities that malefactors may attempt to exploit.

By being smart about the URLs you click on, you can explore the web safely and avoid getting lost in the maze of online threats.

B. Classification Method

a) Random Forest Classification

A well-known machine learning technique, Random Forests (RFs) are praised for their excellent accuracy, resilience, and usability. By utilizing the combined strength of numerous decision trees, this ensemble learning technique offers a number of benefits over single trees.

- Improved Accuracy and Generalizability: RFs efficiently handle the overfitting problem that might afflict single trees by aggregating the predictions of several decision trees trained on various subsets of the data. By using an ensemble technique, the model's generalization skills are enhanced, allowing it to function effectively on non-training data as well.
- Reduced Burden of Feature Selection: RFs do not completely remove the requirement for feature selection, but they are often more resistant to the deleterious effects of superfluous features than single decision trees. Even with very redundant or useless features, performance might still be hampered. Feature selection strategies can increase the interpretability and possibly even the accuracy of the model.
- Inherent Feature significance Analysis: This approach yields insightful feature significance ratings, but it might be difficult to evaluate individual trees within an RF. These scores provide some interpretability despite the ensemble's complexity by quantifying the relative impact of each attribute on the model's predictions.
- Increased Stability: When faced with small alterations in the training data, RFs are more stable than single decision trees. This stability results from averaging several trees' predictions, which lessens the effect of individual tree differences on the final result.

Finally, Random Forests provide a strong and adaptable machine learning instrument. They are a well-liked option for many classification and regression applications due to their accuracy, resilience, and relative simplicity of usage. Techniques like feature significance scores offer important insights on model behavior, even though interpretability might be difficult and feature selection is still critical.

b) Logistic Regression

A close relative of linear regression, logistic regression is essential for machine learning applications requiring categorical predictions. Logistic regression performs well when the outcome is binary (yes/no or 0/1) or falls into several categories, whereas linear regression is best at predicting continuous values.

Here's what sets logistic regression apart:

1. Linear Regression Threshold Problems: A threshold is frequently used in linear regression to translate continuous outputs into binary classifications. But this strategy has two significant drawbacks:

- Outliers: A single data point lying far from the majority (outlier) can significantly skew the best-fit line and, consequently, the threshold, leading to inaccurate classifications.
- Negative Thresholds: Depending on the data, the threshold might fall into negative territory, creating nonsensical interpretations in the context of binary classification (e.g., a negative probability of being spam).

2. The Sigmoid Function: A Mighty Friend: By utilizing the sigmoid function, logistic regression gets beyond these restrictions. The outcome of the linear regression is converted by this mathematical function into a probability between 0 (very improbable) and 1 (very likely). An arbitrary threshold is no longer necessary thanks to its probabilistic output, which also offers a more sophisticated way to understand the model's predictions.

3. Uses and Benefits: Logistic regression has a wide range of uses in different categorization problems, such as:

- Spam filtering: Classifying emails as spam or not spam.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
Article Received:*25 July 2023* ***Revised:****12 September 2023* ***Accepted:****30 September 2023*

- Malware detection: Identifying software as malicious or benign.
- Medical diagnosis: Predicting the presence or absence of a disease based on symptoms.

Logistic regression is a useful tool in many machine learning applications because it provides a reliable and comprehensible method for categorical predictions.

Hypothesis => Z = WX + B

Here Z is the Output (W) is the weight or the intercept and (X) are training data points used. (B) is the error term.

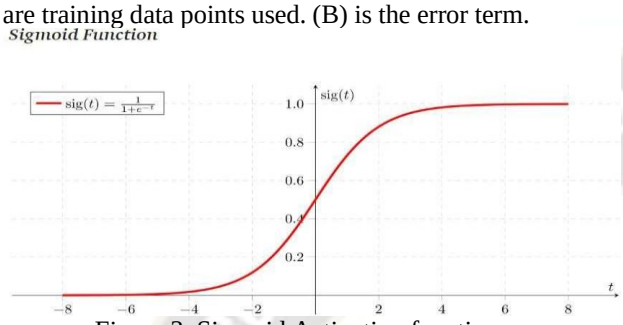

Figure 3. Sigmoid Activation function

In this case, the forecast for "y" will be 1 if "z" moves toward positive infinity and 0 if "z" moves toward negative infinity.

V. Result and Conculsion

Machine learning has replaced signature-based techniques for malware detection, which were limited to known threats. Though machine learning presents an intriguing prospect for identifying undetected viruses, it is not without limitations:

- Black box nature: The model's inability to clearly explain how it makes judgments.
- Dependency on pure, high-quality training data that is susceptible to modification is known as data reliance.

Unlike these drawbacks, machine learning—when paired with additional security measures—represents a substantial advancement in the battle against constantly changing cyberthreats.

VI. Future Enhancemen

In order to improve the effect and user experience of your machine learning-based malware and URL detection project, this study suggests future paths and possible enhancements.

Data-Driven Improvement:

- Dataset Expansion: Although the present dataset focuses on common malware categories, it would be beneficial to include information about a broader range of dangerous software, such as banking malware, spyware, adware, rootkits, and backdoors. The model will be able to recognize a wider variety of threats because to this diversity.
- Enhancing Data Volume: Examine methods to increase the volume of data while maintaining the quality of the data. To obtain access to extensive datasets, cooperate with respected organizations or security research communities.

Expanding Accessibility and Functionality:

- Web-based Deployment: Move the program from a local system to an online platform where users may submit files and URLs for instantaneous examination. More usability and accessibility are promoted by this.

- Interactive Elements:

Put in place features like:

i) Enable simple, user-friendly file submissions for scans with our intuitive file uploader.
ii) Specialized URL Detector: Establish a specific area where users may submit URLs for examination.
iii) Provide unambiguous and enlightening findings that show if a file or URL is considered harmful or benign.

GUI Development: To improve the application's intuitiveness and increase its user base, create a graphical user interface (GUI) that is easy to use for Windows users.

Real-time Integration for Enhanced Security:

- Real-time File Scanning: When downloading or transferring files, incorporate the malware detection model for real-time scanning. This provides an additional degree of protection by seeing possible dangers before they have a chance to do damage.
- Creation of Browser Extensions: Create a browser add-on that serves as a live URL finder. By analyzing the URLs users visit, this plugin may warn users of potentially harmful information.

References

[1] H. Bahassi, N. Edddermoug, A. Mansour and Azmi Mohamed, "Toward an exhaustive review on Machine Learning for Cybersecurity", Procedia Computer Science, vol. 203, pp. 583-587,2022
[2] F. Vitaly and S. Ambareen, "Applications of Machine Learning in Cyber Security. 27th International Conference on Computer Applications in Industry and Engineering", CAINE 2014.
[3] S.Y. Yerima, "High Accuracy Detection of Mobile Malware Using Machine Learning", Electronics 2023, 12, 1408. https://doi.org/10.3390/electronics12061408
[4] V. Vasani, A.K. Bairwa, S. Joshi, A. Pljonkin, M. Kaur and M. Amoon, "Comprehensive Analysis of Advanced Techniques and Vital Tools for Detecting Malware Intrusion", Electronics 2023, 12, 4299. https://doi.org/10.3390/electronics12204299
[5] M. A. Ramirez, S. Yoon, E. Damiani, H. Al Hamadi, C. A. Ardagna, N. Bena, Y. Byon, T. Kim, C. Cho and C.Y. Yeun,"New data poison attacks on machine learning classifiers for mobile exfiltration", in IEEE Access, vol 4, 2016
[6] I. A. Mohammed,"How Artificial Intelligence Is Changing Cyber Security Landscape and Preventing Cyber Attacks: A systematic review",in International Journal of Creative Research Thoughts,Vol. 4, Issue 2 ,June 2016
[7] O. A. Ayeni,"A Supervised Machine Learning Algorithm for Detecting Malware",in Journal of Internet Technology and Secured Transactions,Vol. 10, Issue 1, 2022
[8] R. Patil, W. Deng,"Malware Analysis using Machine Learning and Deep Learning techniques",in IEEE,2021
[9] D. Gavrilut, M. Cimpoesu, D. Anton, L. Ciortuz,"Malware detection using machine learning", in International Multiconference on Computer Science and Information Technology,2009.
[10] L. Duong, D. Cho,"Detecting Malware based on Analyzing Abnormal behaviors of PE File",in International Journal of Advanced Computer Science and Applications, 2021
[11] J. Ali, R. Khan, N. Ahmad, and I. Maqsood, "Random forests and decision trees",in International Journal of Computer Science Issues,2012
[12] A. Shabtai , U. Kanonov, Y. Elovici, C. Glezer, Y. Weiss,"“Andromaly”: a behavioral malware detection",in Journal of Intelligent Information Systems,February 2012