WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Malware Detection and Prevention through Integrated Analysis of Public Search Data

Prathamesh Jadhav[1], Prathamesh Bhavsar[2], Sanket Deore[3], Kiran Kuyate[4], Miss. Gayatri Bendale[5]

[1,2,3,4]BE Students and [5]Asst. Prof. of Department of Computer Engineering,
Matoshri College Of Engineering And Research Centre, Nashik

A B S T R A C T

Malware detection is essential as an early warning system for organizations' security due to the increasing incidence of malware on the Internet. In the proposed work, a unique sentence embedding method is used to link knowledge from separate, specialized datasets. This system suggests a novel method for detecting malware activity that makes use of standard and specialized databases as well as search information from individuals. Actual attack study instances will be used to demonstrate the detection capabilities of our technique. By analyzing the previous system, we saw a rise in searches and model probabilities a few days prior to and following the attack. Additionally, there is an increase in outliers in the time series of the online search volume and model probabilities 14 days before and after the attack is discovered. This approach will prepare the way for the integration of user-generated dynamic data and domain-specific datasets for malware activity detection.

Keywords: Malware Detection, Data Models, Big Data, Transformers, Internet Performance.

I. INTRODUCTION

In the digital age, the Internet has become an integral part of our lives, facilitating communication, commerce, and information exchange on a global scale. However, the widespread connectivity that defines our modern world also exposes us to an escalating threat: malware. Malicious software, or malware, is a pervasive and constantly evolving menace that undermines the security of organizations, individuals, and entire digital ecosystems. The exponential growth in malware incidents has made it imperative for us to develop robust systems capable of early detection and mitigation. In response to this pressing need, we introduce a novel system that aims to revolutionize malware detection. By harnessing the power of advanced data analytics and integrating diverse sources of information, this system offers a proactive and comprehensive approach to identifying and thwarting malware threats.

Central to our system's innovation is the development of a unique sentence embedding method. Traditional malware detection methods often rely on static indicators and signatures, struggling to keep pace with the ever-changing tactics employed by cybercriminals. In contrast, our approach harnesses the power of natural language processing and machine learning to connect knowledge from disparate datasets. This not only enhances the system's ability to detect known malware but also empowers it to identify emerging threats that may not yet have identifiable signatures. By seamlessly linking information from standard and specialized databases, as well as real-time user-generated data, we unlock a new frontier in the fight against malware.

II. LITERATURE REVIEW

The proliferation of malware in the digital landscape has been a subject of extensive research and concern in recent years. As organizations and individuals grapple with increasingly sophisticated cyber threats, researchers and practitioners have been exploring innovative approaches to enhance malware detection and safeguard digital assets. Traditional malware detection methods have largely relied on signature-based techniques, which are effective in identifying known malware but struggle to detect zero-day attacks and polymorphic malware. These limitations have spurred the development of more advanced solutions. Machine learning and deep learning approaches have gained prominence in the field of malware detection. Techniques such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs) have shown promise in improving detection rates by analysing patterns in code and behaviour. However, these approaches often require large, labelled datasets and can be vulnerable to adversarial attacks.

The proposed system distinguishes itself by introducing a novel sentence embedding method, which leverages natural language processing (NLP) and machine learning to connect information from diverse datasets. This approach offers a fresh perspective on malware detection, as it goes beyond code-based analysis and incorporates textual and contextual information. Similar embedding techniques have been successfully applied in various NLP tasks, such as sentiment analysis and document classification, but their application in malware detection is relatively unexplored. A critical aspect of the proposed system is the analysis of historical attack data. Several studies have highlighted the importance of understanding the behaviour and patterns exhibited by malware in previous incidents. This empirical knowledge can serve as a valuable resource for developing more proactive detection models.

Notably, the system's identification of surges in search activity and model probabilities leading up to and following an attack aligns with findings in the broader cybersecurity literature. Researchers have observed that cybercriminals often conduct reconnaissance and information gathering prior to launching attacks, leaving digital footprints that can be detected through careful analysis.

III. SYSTEM ARCHITECTURE

Detecting malware in Portable Executable (PE) files using machine learning algorithms like decision trees, random forests, and Naive Bayes can be an effective approach. Here's a high-level overview of how you can implement such a system:

• **Dataset Collection:** Gather a diverse dataset of both benign and malicious PE files. You can obtain these files from various sources, including open datasets, malware repositories, and clean software installations.

• **Preprocessing**: Clean and preprocess the data. This may involve dealing with missing values, normalizing features, and encoding categorical data.

• **Feature Extraction:** After the preprocessing process was done, features from the dataset were extracted into a feature vector. We used several word n-gram features such as unigram, bigram, trigram, and combination of unigram, bigram, and trigram. Two-term weighting schemes were used for the feature extraction process. The term weighting schemes used were Bag-of-Words (BOW) and Term Frequency Inverse Document Frequency (TF-IDF).

• **Classification:** We implemented several machine learning algorithms as the classifier for target classification of hate speech in tweets. Those algorithms are Support Vector Machine (SVM) and Naive Bayes (NB) According to the previous study of hate speech classification, The training phase used 80% of the dataset as the training data, while the testing phase used the remaining 20% of the dataset as the testing data.

• **Evaluation**: The evaluation measurement used in this study is F1- score. Accuracy is not used as evaluation measurement because it cannot guarantee that high accuracy shows that the model can predict well considering the accuracy paradox. F1- score is obtained by calculating harmonic mean between recission and recall.

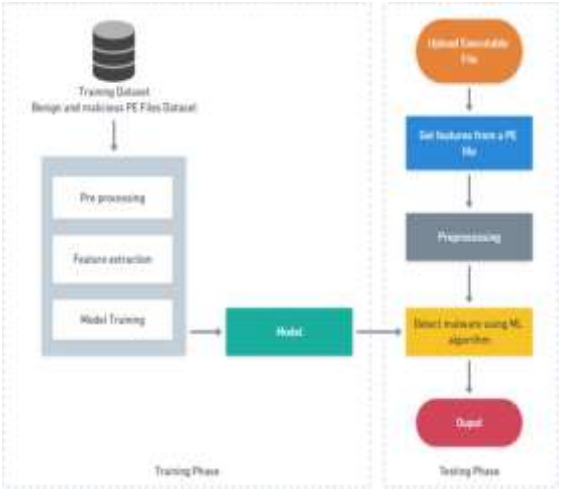

Fig.1: System Architecture

IV. PROJECT IMPLEMENTATION

Detecting malware in Portable Executable (PE) files using machine learning algorithm like decision trees, random forests, and Naive Bayes can be an effective ap- poach. Here's a high-level overview of how you can implement such a system:

- Dataset Collection:

Gather a diverse dataset of both benign and malicious PE files. You can obtain these files from various sources, including open datasets, malware repositories, and clean software installations.

- Preprocessing:

Clean and preprocess the data. This may involve dealing with missing values, normalizing features, and encoding categorical data.

- Feature Extraction:

After the preprocessing process was done, features from the dataset were extracted into a feature vector. We used several word n-gram features such as unigram, bigram, trigram, and combination of unigram, bigram, and trigram. Two-term weighting schemes were used for the feature extraction process. The term weighting schemes used were Bag-of-Words (BOW) and Term Frequency- Inverse Document Frequency (TF-IDF).

V. SEQUENCE DIAGRAM

The purpose of interaction diagrams is to visualize the interactive behaviour of the system. Visualizing the interaction is a difficult task. Hence, the solution is to use different types of models to capture the different aspects of the interaction.

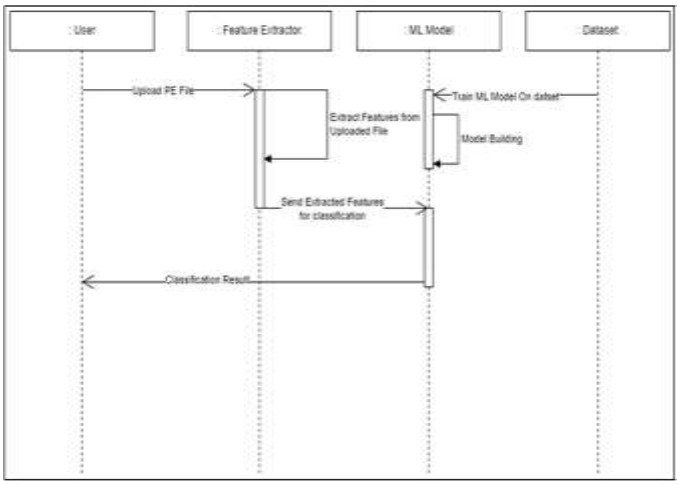

Fig.2: Sequence Diagram

Purpose Of Sequence Diagram:

- To model high-level interaction among active objects within a system.
- To model interaction among objects inside a collaboration realizing a use case.
- It either model's generic interactions or some certain instances of interaction.

The depicted sequence diagram delineates the primary interactions among the user, the system, and the malware detection engine. In a practical scenario, the user uploads a portable executable file and subsequently extracts the necessary features from the file. The machine learning model utilized for malware detection is typically trained on an extensive dataset comprising known malware samples and benign data, enabling it to generate precise predictions.

VI. PROTOTYPE MODEL OF PROJECT

6.1 Sign Up Page:

Fig .3: Sign Up Page

6.2 LOGIN PAGE:

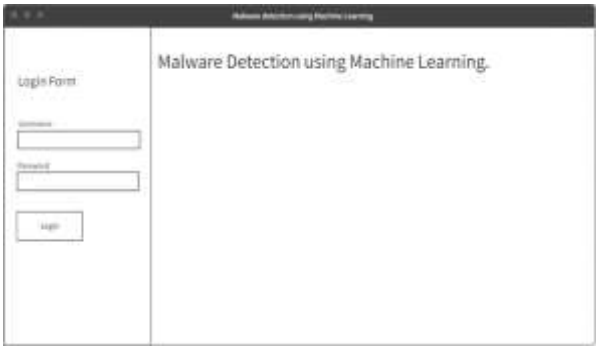

Fig. 4: Login Page

6.3 Main Page:

Fig. 5: Main Page.

VII. CONCLUSION

By automating the analysis process and leveraging advanced feature engineering, the system demonstrates promise in efficiently identifying both known and potentially zero-day threats. However, it is imperative to acknowledge the system's inherent limitations, including the potential for evasion techniques, imbalanced datasets, and the need for ongoing model updates. As the system progresses, addressing these challenges will be pivotal in ensuring its effectiveness and reliability in safeguarding digital environments. With continued refinement, this innovative approach has the potential to become a crucial component of a comprehensive cybersecurity strategy, offering enhanced protection against an ever-evolving landscape of malware threats.

VIII. REFERENCES

B. TAHTACI and B. CANBAY,"Android Malware Detection Using Machine Learning," 2020 Innovations in Intelligent Systems and Applications Conference (ASYU), Istanbul, Turkey, 2020, pp. 1-6, doi: 10.1109/ASYU50717.2020.9259834.

I. Firdausi, C. lim, A. Erwin and A. S. Nugroho," Analysis of Machine learning Techniques Used in Behavior-Based Malware Detection," 2010 Second Interna- tional Conference on Advances in Computing, Control, and Telecommunication Technologies, Jakarta, Indonesia, 2010, pp. 201-203, doi: 10.1109/ACT.2010.33.

A. Irshad, R. Maurya, M. K. Dutta, R. Burget and V. Uher, "Feature Optimization for Run Time Analysis of Malware in Windows Operating System using Machine Learning Approach," 2019 42nd International Conference on Telecommunications and Signal Processing (TSP), Budapest, Hungary, 2019, pp. 255- 260.

K. Sethi, R. Kumar, L. Sethi, P. Bera and P. K. Patra,"A Novel Machine Learning Based Malware Detection and Classification Framework," 2019 International Conference on Cyber Security and Protection of Digital Services (Cyber Security), Oxford, UK, 2019, pp. 1-4, doi: 10.1109/CyberSecPODS. 2019. 8885196.