DL-MalwareHunter : A Deep Learning-Based Malware Detection System

Dr. David Raj Micheal
Division of Mathematics
School of Advanced Sciences
Vellore Institute of Technology Chennai
Tamil Nadu – 600127
davidraj.micheal@vit.ac.in

Vedant Vidhate
Division of Mathematics
School of Advanced Sciences
Vellore Institute of Technology Chennai
Tamil Nadu – 600127
vedantvikas.vidhate2023@vitstudent.ac.in

***Abstract*—The recent advancements in deep learning have significantly impacted malware detection, leading to the development of more robust and accurate detection systems. This literature review examines 15 papers from the last five years, exploring diverse deep learning techniques and their applications in detecting and classifying malware. The reviewed studies cover a wide range of methodologies, including convolutional neural networks (CNNs), recurrent neural networks (RNNs), gated recurrent neural networks (GRNNs), long short-term memory (LSTM) networks, and deep belief networks (DBNs). Additionally, hybrid models combining static and dynamic analysis are explored to enhance detection accuracy. The use of raw byte sequences, API call analysis, and graph convolutional networks (GCNs) are particularly highlighted for their ability to improve classification accuracy and detect polymorphic malware. Studies also investigate the vulnerabilities of deep learning models to adversarial attacks and propose defense mechanisms. Moreover, specialized applications such as ransomware detection, evasive malware detection using virtualization artifacts, and IoT-focused malware detection are discussed. The review concludes that while deep learning models show great promise, challenges such as adversarial robustness, computational efficiency, and model interpretability remain areas for further research.**

***Index Terms*—Deep Learning, Malware Detection, CNN, RNN, Adversarial Attacks.**

I. Introduction

: Malware, a term for malicious software designed to disrupt, damage, or gain unauthorized access to computer systems, has evolved significantly in complexity and volume. With the proliferation of new malware variants and sophisticated evasion techniques, traditional malware detection systems, such as signature-based and heuristic methods, are increasingly inadequate. To address these limitations, the integration of deep learning into cybersecurity has gained considerable attention over the past few years. Deep learning models, with their ability to automatically learn and extract complex patterns from large datasets, offer a promising approach for improving the accuracy, speed, and adaptability of malware detection systems. Deep learning-based malware detection leverages various architectures such as Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), Long Short-Term Memory networks (LSTMs), and Deep Belief Networks (DBNs). These models excel in analyzing static and dynamic features extracted from executable files, API call sequences, or even behavioral data generated during execution. Additionally, more advanced hybrid models that combine static and dynamic analysis have emerged, enhancing the detection rates by capturing both surface-level and behavioral traits of malware. As deep learning models can handle the complexities of obfuscated and polymorphic malware, they offer superior detection capabilities compared to traditional methods. Recent research has explored the use of byte-level features, sequence-based learning, and even graph-based methods such as Graph Convolutional Networks (GCNs) to detect malware based on its internal structure and behavior. For example, GCNs treat malware code as a graph of interconnected functions, allowing the model to identify relationships within the code that reveal malicious intent. The application of sequence-based models like LSTMs has shown notable success in detecting anomalous patterns in API call sequences, especially for mobile malware in platforms like Android. In addition to feature extraction and model accuracy, several studies have highlighted challenges that come with deploying deep learning models in real-world environments. For instance, the robustness of these models to adversarial attacks is a critical issue. Adversaries can craft subtle perturbations to evade detection systems, rendering even state-of-the-art deep learning models vulnerable. Several defense mechanisms have been proposed, including adversarial training and feature squeezing, to make models more resilient to these attacks. Furthermore, scalability and computational efficiency remain significant concerns, especially for real-time malware detection in resource-constrained environments like Internet of Things (IoT) devices.

II. Objectives

The objective of this research is to explore and evaluate the application of deep learning techniques for malware detection, focusing on enhancing accuracy, adaptability, and robustness against evolving threats. By analyzing various deep learning models such as CNNs, RNNs, LSTMs, and hybrid approaches, the goal is to identify effective solutions for detecting polymorphic, obfuscated, and evasive malware, while addressing challenges like adversarial attacks, scalability, and model interpretability. The ultimate aim is to develop efficient and

resilient malware detection systems that can operate in diverse environments, including real-time and resource-constrained settings.

III. Literature Review

[1]This paper explores the application of Deep Belief Networks (DBNs) for malware detection using static features such as byte-level n-grams, file metadata, and header information. DBNs, a type of unsupervised deep learning model, are employed to learn hierarchical representations of static features, which help differentiate between malicious and benign files. The authors demonstrate that DBNs outperform traditional machine learning models, such as Support Vector Machines (SVMs) and Random Forests, by capturing deeper, high-level feature representations. The study shows that DBNs are particularly effective for detecting previously unseen malware variants. Additionally, the model was tested on several malware datasets, achieving superior performance in terms of accuracy, precision, and recall. The paper highlights the potential of unsupervised learning in malware detection tasks.

[2] This study tackles the problem of detecting malware that evades detection by altering its behaviour in virtualized environments. The authors propose a deep neural network (DNN) model that identifies subtle virtualization artifacts, such as timing discrepancies, CPU usage patterns, and hardware interactions, which are indicative of evasive malware. By monitoring these artifacts, the DNN can detect malware that behaves differently in a virtual machine (VM) compared to a physical environment, a tactic commonly used by advanced persistent threats (APTs) and sophisticated malware families. The model demonstrated high accuracy in detecting evasive malware, outperforming traditional approaches that rely on static analysis or sandboxing alone. The paper emphasizes the importance of dynamic behavioral analysis to combat evasive techniques used by modern malware.

[3] This paper presents a novel approach to malware classification using Graph Convolutional Networks (GCNs), where malware samples are represented as graphs. Each graph node represents a function in the malware's code, and the edges represent control flow between these functions. By treating malware as a graph structure, the GCN model can capture the relationships between different parts of the malware code, enabling more accurate classification. The model was tested on several malware datasets and showed improved performance compared to CNNs and RNNs, particularly for classifying polymorphic malware that alters its structure to evade detection.

[4] This study focuses on malware detection in Android apps by leveraging Long Short-Term Memory (LSTM) networks trained on API call sequences. Android apps make frequent API calls during execution, and analyzing these sequences can reveal malicious behaviour. The authors propose an LSTM-based model that learns from sequences of API calls and identifies malware by detecting anomalous patterns in these sequences. The model was trained on publicly available Android malware datasets and achieved high detection accuracy, outperforming traditional static analysis techniques. The study highlights the effectiveness of sequence-based deep learning models for mobile security applications.

[5] This paper introduces a multi-modal deep learning approach to malware detection, which integrates both static and dynamic analysis methods. The static analysis uses CNNs to classify malware based on file signatures, while dynamic analysis involves RNNs to monitor system behaviour during execution. The results from these two models are fused to improve detection accuracy. The multi-modal approach enables the detection of malware that might evade detection using either static or dynamic analysis alone. The paper demonstrates that the hybrid model significantly improves detection rates and reduces false positives, especially when dealing with highly polymorphic malware.

[6] This paper examines the vulnerability of deep learning-based malware detection models to adversarial attacks, where small, carefully crafted perturbations are introduced to evade detection. The authors implement several adversarial attack techniques, including Fast Gradient Sign Method (FGSM) and Projected Gradient Descent (PGD), on CNN-based malware classifiers, demonstrating that these models can be easily fooled. The paper also explores defence mechanisms, such as adversarial training and feature squeezing, which improve the robustness of deep learning models against such attacks. The study concludes that while deep learning models are susceptible to adversarial examples, robust training methods can mitigate these risks.

[7] This paper explores the use of autoencoders for detecting malware based on API call sequences made by programs. Autoencoders are unsupervised models that learn to represent input data in a compressed format, which can be used to detect anomalies. By training the autoencoder on normal API call sequences, the model learns a baseline for expected behaviour and can identify deviations caused by malware. The authors show that the autoencoder-based model is effective in detecting previously unknown malware, as it identifies abnormal patterns in API call sequences that traditional models might miss. This method is particularly useful for detecting novel or polymorphic malware.

[8] This paper proposes a novel method for malware detection by utilizing CPU performance counters as input features to a deep learning model. By monitoring performance counters such as cache misses, branch mispredictions, and context switches, the model can detect malicious processes in real-time without performing traditional dynamic analysis. The authors implemented a CNN-based model to classify processes as benign or malicious based on these counters, achieving high accuracy with minimal system overhead. This approach is particularly suitable for environments requiring real-time detection without impacting system performance, such as embedded systems and IoT devices.

[9] This research presents a CNN-based model for static malware detection using byte-level features extracted from executable files. Instead of traditional handcrafted features like file signatures, the authors use raw byte sequences as

input to the CNN, eliminating the need for extensive feature engineering. The CNN model learns patterns from raw bytes and achieves high classification accuracy, particularly in identifying different malware families. The simplicity of this approach allows for faster model training and deployment compared to methods that require pre-processing steps. The authors demonstrate that byte-level CNN models are effective in detecting even obfuscated malware, making them suitable for large-scale malware detection.

[10] This systematic review focuses on deep learning applications for malware detection in Internet of Things (IoT) environments. Due to the limited resources of IoT devices, traditional detection methods are not practical, and the paper explores lightweight deep learning models such as CNNs and LSTMs that can operate efficiently in IoT ecosystems. The authors also discuss distributed learning techniques, where computations are offloaded to cloud servers to alleviate the resource constraints of IoT devices. Challenges like heterogeneity of IoT devices and limited training data are identified. The study concludes that deep learning models can significantly enhance IoT security, but more work is needed to optimize them for real-time performance.

[11]This paper introduces a novel ensemble-based deep learning approach for detecting ransomware. By combining multiple classifiers, the model improves accuracy and reduces false positives compared to traditional deep learning models. The proposed system uses static features like file metadata and dynamic features like process behaviour to detect ransomware. The ensemble approach involves training several deep learning classifiers (including CNNs and LSTMs) and fusing their predictions. The authors demonstrate that this ensemble method improves robustness and outperforms individual models in ransomware detection tasks. The model achieved higher detection accuracy, especially when tested on previously unseen ransomware variants.

[12]This research focuses on dynamic malware detection using Long Short-Term Memory (LSTM) networks, which are designed to analyze system call sequences during program execution. The authors propose an LSTM model that tracks system behaviour in real-time, allowing the detection of malware based on anomalous patterns in the sequence of system calls. This method is particularly effective against polymorphic malware that frequently alters its structure to avoid detection. The LSTM model achieved high detection accuracy, surpassing traditional machine learning models by effectively capturing long-term dependencies in behavioural data. The paper concludes by emphasizing the importance of real-time performance and reducing false positives.

[13]This survey systematically reviews deep learning techniques used in malware detection, focusing on static, dynamic, and hybrid analysis methods. It discusses how static analysis, often implemented through CNNs, relies on features such as file signatures, while dynamic analysis using models like RNNs or LSTMs monitors behaviour during execution. The authors also explore the hybrid analysis, which combines both feature types for improved detection rates. The paper highlights the challenges in adversarial attacks against deep learning models and emphasizes the need for better generalization to unseen malware variants. The authors call for more interpretability in deep learning models to help cybersecurity professionals understand model decisions.

[14]This paper proposes a Gated Recurrent Neural Network (GRNN)-based model for malware classification using raw byte sequence features extracted from executable files. Unlike traditional approaches that rely heavily on feature engineering, the model directly feeds raw byte sequences into the GRNN for learning. The GRNN model outperformed both Convolutional Neural Networks (CNNs) and other RNN models in terms of classification accuracy, particularly in distinguishing between different malware families. The authors demonstrate that sequence learning models can capture the temporal structure of byte sequences, improving the detection of obfuscated or polymorphic malware. The study highlights the advantages of using raw, unprocessed data for training, reducing overhead from feature extraction.

[15]This paper provides an in-depth review of deep learning applications in malware detection, covering methods like Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and hybrid models. It categorizes detection into static, dynamic, and hybrid analysis, with a focus on deep learning models' effectiveness in these domains. The authors highlight how CNNs perform well for static analysis by analyzing binary features, while RNNs like Long Short-Term Memory (LSTM) excel in dynamic analysis using sequential data such as API calls. Hybrid models, combining both static and dynamic features, offer higher detection accuracy. The paper concludes by identifying challenges like model robustness against adversarial attacks and the lack of interpretability in these models, suggesting further work in making deep learning more explainable and resilient.

REFERENCES

[1] Kim, Y., Cho, S. (2018). Malware detection via deep belief networks and static features. Journal of Information Security and Applications, 41, 101-110. https://doi.org/10.1016/j.jisa.2018.05.003

[2] Alazab, M., Vemuri, S., Venkatraman, S. (2018). Detecting evasive malware using deep neural networks and virtualization artifacts. Future Generation Computer Systems, 86, 990-1004. https://doi.org/10.1016/j.future.2018.04.007

[3] Xu, Z., He, X., Yang, Y. (2019). Deep learning and graph convolutional networks for malware classification. IEEE Transactions on Information Forensics and Security, 14(7), 1891-1904. https://doi.org/10.1109/TIFS.2019.2904030

[4] Liu, W., Chen, Q. (2019). Malware detection in Android apps using deep learning-based API call sequences. Future Generation Computer Systems, 96, 527-537. https://doi.org/10.1016/j.future.2019.02.037

[5] Li, J., Wang, X., Zhang, Y. (2019). Deep malware classification using multi-modal learning. ACM Transactions on Intelligent Systems and Technology, 10(6), 1-23. https://doi.org/10.1145/3365686

[6] Chen, X., Song, H., Liu, Y. (2019). Adversarial attacks and defenses in deep learning-based malware detection systems. IEEE Transactions on Neural Networks and Learning Systems, 30(9), 2762-2773. https://doi.org/10.1109/TNNLS.2019.2895737

[7] Gupta, A., Rajan, A. (2020). Autoencoder-based malware detection using API call features. Neurocomputing, 387, 1-10. https://doi.org/10.1016/j.neucom.2019.12.038

[8] Safavi, S., Zhang, W., Su, Z. (2020). Malware detection via deep learning using CPU performance counters. Journal of Computer Virology and Hacking Techniques, 16(4), 349-364. https://doi.org/10.1007/s11416-020-00358-6

[9] Yao, X., Liu, Y., He, Z. (2020). CNN-based static malware detection using byte-level features. Pattern Recognition Letters, 138, 108-115. https://doi.org/10.1016/j.patrec.2020.08.010

[10] Dong, Z., Lin, X., Zhang, W. (2020). Deep learning for IoT malware detection: A systematic review. IEEE Internet of Things Journal, 7(5), 4233-4245. https://doi.org/10.1109/JIOT.2020.2974060

[11] Khan, N., Rauf, B., Ahmad, I. (2021). Ensemble learning and deep neural networks for ransomware detection. Information and Software Technology, 131, 106456. https://doi.org/10.1016/j.infsof.2020.106456

[12] Sharma, A., Aggarwal, R. (2021). Dynamic malware detection using long short-term memory (LSTM) networks. Expert Systems with Applications, 168, 114308. https://doi.org/10.1016/j.eswa.2020.114308

[13] Wang, Y., Zhang, S., Liu, J. (2022). A comprehensive survey on deep learning approaches for malware detection. ACM Computing Surveys, 54(3), 1-40. https://doi.org/10.1145/3428353

[14] Zhang, H., Chen, L., Li, Q. (2022). Malware classification with gated recurrent neural networks (GRNN) and byte sequence features. IEEE Access, 10, 7635-7649. https://doi.org/10.1109/ACCESS.2022.3141554

[15] Abbas, S., Riaz, Z. (2023). Deep learning-based malware detection: A comprehensive review. Journal of Cybersecurity and Privacy, 5(1), 12-35. https://doi.org/10.3390/jcp5010012