International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 06 | June - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

Malware Detection Using Machine Learning and Deep Learning

Mrs. SUGUNA MK ASSISTANT.PROFFSOR OF SMVIT
POOJA RAI ,STUDENT ,SMVIT,
PRIYA KUMARI ,STUDENT ,SMVIT,
NISHTHA SHANDILYA , STUDENT ,SMVIT,
WASIL AHMAD , STUDENT , SMVIT

Abstract—With the rise of polymorphic and zero-day malware, traditional signature-based detection methods have become insufficient. This paper proposes a hybrid malware detection framework that integrates machine learning (ML) and deep learning (DL) with dynamic behavioral analysis. By utilizing the CIC-MalMem-2022 dataset and extracting runtime features such as API calls and memory usage, the framework applies models like CNN, LSTM, and Decision Trees. Experimental results show that our approach achieves up to 99.27% accuracy while reducing false positives. This demonstrates the potential of intelligent, real-time, and scalable malware detection systems integrated with hardware-assisted techniques.

1. INTRODUCTION

With the rapid advancement of digital technology, cyberattacks have become more frequent, complex, and damaging—affecting everything from personal devices to critical national infrastructure. Among these threats, malware (including viruses, ransomware, and spyware) remains one of the most persistent and evolving forms of cybercrime. As malware becomes more sophisticated, it increasingly bypasses traditional security tools, posing a serious risk to individuals, organizations, and governments alike.

The rise of interconnected devices—smartphones, IoT gadgets, and cloud servers—has expanded the attack surface significantly. Each platform comes with its own set of vulnerabilities, making universal protection more challenging. Real-world incidents like the Killnet cyberattacks on U.S. state services in 2022 and the theft of COVID-19 relief funds by Chinese-linked threat actors underline the urgent need for more proactive and intelligent detection systems.

To meet this challenge, researchers are turning to deep learning (DL) techniques such as Convolutional Neural Networks (CNNs), Deep Neural Networks (DNNs), and Recurrent Neural Networks (RNNs). These models excel at detecting complex patterns and have shown high accuracy in identifying even new, unknown, or polymorphic malware that traditional methods often miss.

At its core, malware detection aims to identify and neutralize malicious software before it can damage systems or steal sensitive data. The earlier malware is detected, the better the chance to prevent harm and respond in real time. Modern
detection approaches fall into several key categories:

Signature-Based Detection:

This method identifies malware by comparing files against a database of known malware signatures. It is efficient for detecting previously seen threats but lacks the ability to identify new or polymorphic malware variants.

Heuristic-Based Detection:

Heuristic methods apply rule-based algorithms to identify anomalous code patterns or behaviors. While effective at detecting modified or slightly altered malware, they tend to generate false positives.

Behavior-Based Detection:

This approach monitors the dynamic behavior of programs—such as system calls, memory access, and process execution patterns—to identify threats. It is effective against zero-day and polymorphic malware, though it demands higher computational resources.

Machine Learning-Based Detection:

Leveraging data-driven algorithms, this technique involves training models on features extracted from known benign and malicious samples. These models adapt to evolving threats and offer high detection accuracy, but they require extensive, well-labeled datasets and periodic retraining.

Hardware-Assisted Detection:
This emerging technique uses low-level system data such as CPU instruction traces and memory access patterns to detect malware. It is difficult for attackers to evade but typically requires integration with specialized hardware platforms.

Hackers aim to install malware on a victim's computer, often bypassing firewalls by tricking users into running malicious code. A common method is sending phishing emails with infected documents or links. Once opened, these documents execute hidden scripts that download the actual malware—such as ransomware or backdoors. These malicious files are typically just the entry point, not the final payload. For example, a PDF file can be used to launch such an attack.

Role of Deep Learning in Modern Malware Detection
As cyber threats become increasingly evasive and polymorphic, traditional malware detection techniques—such as signature-based and rule-based systems—struggle to maintain accuracy and relevance. In contrast, deep learning (DL), a subfield of machine learning inspired by the human brain's neural networks, offers powerful mechanisms for identifying complex patterns in large datasets. DL models have proven particularly effective in malware detection by learning directly from raw input data, eliminating the need for extensive manual feature engineering. Two of the most impactful architectures in this space are Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), including their variants such as Long Short-Term Memory (LSTM) networks.

Convolutional Neural Networks (CNNs)
CNNs are widely used in image classification tasks and have been successfully adapted for malware detection by treating malware binaries as images. In this approach, executable files (e.g., PE files) are transformed into grayscale images where each byte of the file represents a pixel intensity. CNNs can then detect spatial patterns and structural similarities that may indicate malicious behavior.

Malware Detection Using Machine Learning

Malware detection typically relies on two main approaches: signature-based detection, which identifies known threats using predefined patterns, and behaviour-based detection, which focuses on suspicious activity. Underlying these are two key analysis methods: static analysis and dynamic analysis. Static analysis examines a file without running it—much like reading its blueprint—to uncover clues through metadata, embedded strings, file structure, or disassembled code. It's thorough but can miss hidden or obfuscated threats. In contrast, dynamic analysis runs the malware in a safe, controlled environment to observe real-time behaviour such as system changes, network access, or API calls. This provides valuable insight into how the malware actually operates, making it especially useful for detecting stealthy or new variants. Machine learning enhances both methods by learning to recognize patterns in these behaviours, improving the detection of even previously unknown malware.

How Machine Learning Enhances Malware Detection

Machine learning (ML) has revolutionized antivirus software by enabling it to detect both known and emerging malware—like zero-day and polymorphic threats—that traditional signature-based methods often miss. By learning patterns from large datasets, including system behaviour and file characteristics, ML models such as decision trees, SVMs, and neural networks can spot suspicious activity in real time.

As new threats appear, these models adapt and improve, making detection smarter over time. While hybrid approaches using static and dynamic analysis with deep learning have shown high accuracy, challenges like false positives, limited scalability, and lack of transparency remain. To tackle these, our research introduces a robust framework that uses feature selection, transfer learning, and Explainable AI (XAI) to make malware detection more accurate, adaptable, and easier to trust in real-world use.

2. LITERATURE SURVEY

Aim:

[1] The primary goal of this research is to develop an intelligent and adaptive malware detection system that overcomes the limitations of traditional signature-based methods. It focuses on using a hybrid approach that combines machine learning (ML) and deep learning (DL) with dynamic behavioral analysis (like memory access patterns and API calls) to detect advanced threats such as polymorphic and zero-day malware.

Results:

Decision Tree (DT) model achieved 99.27% accuracy, the highest among all models tested.

CNNs and LSTMs from deep learning also performed strongly, effectively identifying malware by learning patterns from grayscale binary images and sequential API calls.

The use of dynamic datasets and feature selection techniques (RFE, correlation filtering) significantly improved performance and reduced false positives.

The proposed system proved to be more scalable, accurate, and robust compared to traditional methods, especially in detecting unseen or obfuscated malware.

Aim:

[2] This paper explores the increasing threat of malware and proposes a deep learning-based approach to improve malware detection. It highlights the challenge of handling high-dimensional data and leverages correlation-based feature selection* to improve efficiency and reduce computation in detection systems.

To build a robust, accurate, and low-computation malware detection system using:

- Deep learning models (Dense & LSTM)
- Correlation-based feature selection
- Two datasets with different characteristics (one with many records, one with many features)

Results:

- For the first dataset (with 33 features), the Dense model achieved 100% accuracy with all features and 99.79% even with a 36.63% reduction in features.
- For the second dataset (with 214 features), using only 18.22% of features led to a small drop: from 98.93% to 95.34% accuracy.
- Adding LSTM layers gave minor accuracy improvements but increased training time.
- Feature selection effectively reduced data size while maintaining high detection accuracy, demonstrating the model's efficiency and reliability.

Aim:

[3] This paper explores the increasing threat of malware and proposes a deep learning-based approach to improve malware detection. It highlights the challenge of handling high-dimensional data and leverages correlation-based feature selection to improve efficiency and reduce computation in detection systems.

To build a robust, accurate, and low-computation malware detection system using:

- Deep learning models (Dense & LSTM)
- Correlation-based feature selection
- Two datasets with different characteristics (one with many records, one with many features).

Results:

- For the first dataset (with 33 features), the Dense model achieved 100% accuracy with all features and 99.79% even with a 36.63% reduction in features.
- For the second dataset (with 214 features), using only 18.22% of features led to a small drop: from 98.93% to 95.34% accuracy.
- Adding LSTM layers gave minor accuracy improvements but increased training time.
- Feature selection effectively reduced data size while maintaining high detection accuracy, demonstrating the model's efficiency and reliability.

Aim:

[4] Malware remains a serious and growing threat to computer systems, exploiting the open nature of networks and technologies. Despite various detection technologies like antivirus, firewalls, and encryption, malware continues to evolve, becoming more sophisticated and harder to detect. This paper analyzes current detection techniques and highlights their limitations.

To review and analyze the latest malware detection techniques, categorize them (e.g., signature-based, behavior-based, heuristic, etc.), and identify current issues and challenges in malware detection to inform future research directions.

Results:

- Various detection methods exist, but each has its own limitations, such as high false positives, reliance on signature databases, and poor scalability.
- Emerging techniques like machine learning and hybrid models show promise but still face challenges like feature selection, evasion techniques, and computational overhead.
- There's a need to reduce dependency on large signature databases and develop lightweight solutions suitable for devices like IoT sensors.

Aim:

To build a deep learning-based malware detection system that analyses API call sequences by converting them into grayscale images and combining CNN, TextCNN, and Bi-LSTM models—enhanced with attention mechanisms and transfer learning—for improved accuracy.

Result:

The model achieved over 99% accuracy in both binary and multi-class detection, outperforming earlier methods. By blending visual and semantic features of API behavior, it proved highly accurate, scalable, and effective against both known and unknown malware threats.

Aim:
To develop a scalable and accurate machine learning framework for malware detection using a cascade of perceptron-based classifiers, designed to handle large datasets while minimizing false positives through smart feature selection.
Result:
The proposed framework achieved up to 98% accuracy and proved effective on large datasets. It maintained strong performance with low false positives, confirming that perceptron-based ML models can significantly enhance malware detection and support traditional antivirus systems.
significantly improve malware detection and complement traditional antivirus systems.

Aim:
To develop an automated and lightweight malware detection system using a hybrid deep learning model (CNN + BiLSTM) that analyses grayscale images of malware binaries, enabling accurate detection without manual feature extraction or unpacking.
Result:
The CNN–BiLSTM model effectively detects both known and modified malware variants by analysing binary files as images and capturing sequential patterns. It offers high detection performance with reduced computational overhead, making it suitable for real-time use.

Aim:
To present Kaspersky Lab's multi-layered ML-based malware detection framework that uses supervised and unsupervised learning, similarity hashing, and deep neural networks to detect known and unknown threats with high accuracy, speed, and interpretability.
Result:
The system effectively detects polymorphic and rare malware by combining lightweight pre-scans with deep learning-based post-analysis. It reduces false positives, adapts to new threats, and deploys efficiently through model distillation for real-world use.
-
Aim:
[9] The paper aims to conduct a comprehensive systematic literature review (SLR) of machine learning algorithms used in malware detection. It builds a taxonomy of existing approaches, identifies current challenges, evaluates algorithm performance, and provides future research directions.
Result:
- Support Vector Machine (SVM), Decision Tree (DT), and N-gram approaches showed high accuracy in detecting malware (up to 100% in small datasets).
- An empirical study using a large dataset (EMBER 2018) showed slightly reduced accuracy:
  - SVM: 98.62%, DT: 96.49%, N-gram: 97.43%.
- The size and quality of the dataset significantly impact detection accuracy.
- Behavior-based classification and dynamic/hybrid analysis methods were found more effective than static or signature-based techniques.
- Key challenges include dataset limitations, obfuscated malware, false positives, and analysis type constraints.

Aim:
[10] To build a deep learning model that can detect and classify IoT malware using byte sequences from ELF files.
IoT devices are growing fast but often lack security, making them easy targets for malware. This study uses a deep learning model (Bi-GRU-CNN) to detect malware across different device types without complex feature extraction.
:**Result:**
The Bi-GRU-CNN model effectively detects and classifies IoT malware by analyzing raw byte sequences from ELF files. It works across various device types without needing complex feature engineering, making it a practical and efficient solution for IoT security.

Aim :
[11] To detect malware using deep learning combined with correlation-based feature selection for better performance and reduced complexity.
Malware is a growing cyber threat, especially with the rise of mobile and IoT devices. This study uses deep learning and feature selection to build an efficient malware detection system that works well even with large or complex data.
Result:
The proposed deep learning model, enhanced with correlation-based feature selection, achieved high detection accuracy while significantly reducing data complexity. It performed efficiently on large and complex datasets, proving effective for real-world malware detection, especially in mobile and IoT environments.

Aim:
[12]To review recent deep learning approaches for malware and intrusion detection across platforms like Android, Windows, and IoT.
Result:
The review found that deep learning techniques significantly enhance detection accuracy and adaptability across diverse systems, offering a strong alternative to traditional methods but still facing challenges like explainability and data quality.

Aim:
[13] To build a deep learning model that can automatically detect and classify Android malware using static analysis.
With Android being the most used mobile system, malware targeting it is growing fast. This paper proposes a deep learning method using key app features like permissions and API calls to detect and classify malware accurately without running the apps.

Aim :
[14] To detect Android malware using a deep learning model (GRU) based on app permissions and API calls.
Android is widely used and targeted by malware. This paper uses static features and a GRU-based deep learning model to detect malware more effectively than traditional methods.
Result:
The GRU-based deep learning model successfully detected Android malware using static features like permissions and API calls. It outperformed traditional methods by accurately identifying threats with minimal feature engineering, making it a practical solution for mobile security.

Aim :
[15] To detect malware using machine learning and deep learning models based on opcode frequency features.
Malware is increasing fast and traditional antivirus struggles to keep up. This paper uses opcode patterns and models like Random Forest and Deep Neural Networks to classify malware effectively.
Result:
The survey highlighted that deep learning models—especially CNNs and transfer learning—are highly effective for malware detection. However, it also emphasized ongoing issues such as the need for better datasets, improved model transparency (XAI), and stronger resistance to adversarial attacks, guiding future research directions.

3. PROPOSED METHODOLOGY

To address the limitations of traditional signature-based malware detection—especially in identifying obfuscated or zero-day threats—this study proposes a machine learning-based framework that leverages dynamic behavioral features for enhanced detection accuracy.

1. Addressing the Limitations of Traditional Techniques

Traditional antivirus tools often miss new or disguised malware because they rely on known signatures. To address this, we used a machine learning approach that learns from behavioural patterns—

like CPU usage and API calls—captured in the **CIC-MalMem-2022** dataset. This allows the system to detect even unseen or obfuscated threats by analysing how malware behaves at runtime.

2. Filling the Gap: Dynamic Behavioral Features in ML Models
Traditional methods often miss sophisticated malware due to their static nature. To overcome this, we used the CIC-MalMem-2022 dataset, which provides rich behavioural data like CPU usage, memory access, and API calls—signals that are difficult for malware to hide.
algorithms: Random Forest, Decision Tree, SVM, KNN, and Naïve Bayes-allowing for a broader comparison of performance.

3. Designing a Reproducible, Validated Approach
To assure reliability and scientific grounding of our approach:
We performed a train-test split (70:30) with stratified sampling to preserve the original class distribution.
5-fold cross-validation was used to validate the generalization of the model and suppress overfitting.
Well-accepted evaluation metrics (Accuracy, Precision, Recall, F1-score) were used for the measurement of model performance, and confusion matrices were used to visualize misclassifications.

Preprocessing and Dataset Collection:
Dataset Utilized:
The author utilized the CIC-MalMem-2022 dataset. It has static and dynamic analysis feature of malware samples belonging to various families.

Preprocessing Operations:
Missing Value Handling: Missing value rows were removed.
Label Encoding: Malware and benign labels were converted to numerical representation.
Feature Normalization: Min-max normalization was used to normalize feature value between 0 and 1.
Machine Learning Algorithms Used:
Standard Models:
Random Forest (RF)
Decision Tree (DT)
Support Vector Machine (SVM)
K-Nearest Neighbors (KNN)
Naïve Bayes (NB)
Experimental Tools and Environment
Platform Used:
Python (Jupyter Notebook)
Libraries/Packages:
pandas, scikit-learn, matplotlib, seaborn
Environment: Jupyter Notebook, Standard Desktop, Intel i5, 8GB RAM.

Model	Accuracy	Precision	Recall	F1-score
Decision Tree	99.27%	99.26%	99.28%	99.27%
Random Forest	98.64%	98.60%	98.65%	98.62%
SVM	97.45%	97.74%	97.72%	97.73%
Naïve Bayes	94.87%	95.42%	95.45%	94.67%

TABLE 1.0

Hardware:
Typical desktop machine with Intel is processor and 8GB RAM.
These techniques lower overfitting, enhance accuracy, and reduce computation time by choosing the most important feature

Use of Dynamic Behavioural Dataset (CIC-MalMem-2022)
The CIC-MalMem-2022 dataset captures real-time malware behaviour—like CPU usage and API calls—which static datasets miss. Such dynamic features are harder to obfuscate, making them ideal for detecting zero-day and polymorphic malware. This approach is widely supported in research for its resilience against code evasion techniques.
v malware).
The F1 score is the balancing factor.

4. RESULT AND DISCUSSION

This study assessed the effectiveness of various machine learning (ML) and deep learning (DL) algorithms for malware detection by using both static and dynamic analysis techniques. The classification process was divided into two main phases: training and testing. During the training phase, each algorithm was fed labelled datasets containing both malicious and benign files. Using learning algorithms, the classifiers—including Decision Trees (DT), k-Nearest Neighbours (KNN), Convolutional Neural Networks (CNN), Naive Bayes (NB), Random Forest (RF), Support Vector Machines (SVM), and Logistic Regression—learned to differentiate malware from clean files. In the testing phase, these models were evaluated on new, previously unseen files to assess their accuracy and reliability in real-world scenarios.

The results revealed that DL models, particularly CNNs and Long Short-Term Memory (LSTM) networks, exhibited superior performance compared to traditional ML approaches. CNNs achieved strong results when malware binaries were transformed into grayscale images, extracting spatial hierarchies of patterns effectively. LSTMs excelled in analyzing sequential data such as API call logs, capturing temporal dependencies in malicious behaviors.

Among the ML classifiers, Decision Trees emerged as the most accurate, achieving a classification accuracy of 99%, a true positive rate (TPR) of 99.07%, and a false positive rate (FPR) as low as 2.01%, as shown in Figure 5. These metrics were significantly higher than those of KNN, NB, SVM, RF, and even Logistic Regression. CNNs also performed well, reaching an average accuracy of 96.7% with an FPR of only 2.8%. Such performance underscores the ability of DL models to autonomously learn complex patterns without requiring extensive manual feature engineering.

These findings confirm that while traditional ML methods are still valuable, especially when trained on engineered static features like PE headers and opcode sequences, deep learning offers a more scalable and robust solution for detecting evolving malware threats. Figure X further visualizes the comparative accuracy and reliability of all classifiers evaluated in the study.

Feature Selection Outcomes

- After preprocessing, **RFE and correlation filtering** were used to select the **top 20 most relevant features**, resulting in a more efficient and less overfitted model.
- Among the tested models, **Random Forest performed best**, with **98.64% accuracy** and strong precision, recall, and F1-score.
- **Decision Tree** and **SVM** also showed high performance, though slightly lower.
- **Naïve Bayes** underperformed due to its assumption of feature independence, which doesn't suit the behavioural nature of the dataset.

Comparison with Previous Studies
The findings of this study are consistent with earlier research

demonstrating the power of deep learning (DL) in malware detection. Studies by Hardy et al. (2016) and Tobiyama et al. (2017) showed how models like CNNs and LSTMs could accurately detect malware by analysing file structures and behavioural patterns. These foundational works helped shape the use of image-based and sequence-based DL techniques in cybersecurity.

Later research by Xie et al. (2020) and Liu et al. (2019) extended these ideas, showing that combining static and dynamic analysis with CNNs significantly boosts detection accuracy—especially for Android malware. Gupta et al. (2021) further explored DL's limitations, revealing vulnerabilities to adversarial attacks and underscoring the need for explainable AI.

Our study builds on these insights, confirming the effectiveness of CNNs and LSTMs, while also addressing critical gaps such as model explainability, scalability, and performance on dynamic behavioural datasets.

International Journal of Security and Its Applications, vol. 13, no. 2, pp. 93–100, 2019.

[5] M. Gupta, R. Gupta, and C. Lal, "Adversarial Attacks and Defenses over Deep Learning Models in Malware Classification," *Computers & Security*, vol. 105, p. 102224, 2021. doi: 10.1016/j.cose.2021.102224.

5.Summary and Future Impact

This study reinforces the growing body of research that shows deep learning (DL)—especially CNNs and LSTMs—offers powerful tools for detecting modern malware, even when traditional signature-based methods fall short. By integrating both static features (like file structures) and dynamic behaviours (such as API call sequences), our detection pipeline is better equipped to handle obfuscated and zero-day threats. Models like CNNs performed exceptionally well in classifying malware images, while LSTMs captured sequential patterns in behavioural data with high precision. Compared to earlier approaches (Hardy et al., Tobiyama et al., Liu et al., Xie et al.), our work shows greater scalability, adaptability, and robustness, thanks to the use of dynamic datasets, feature selection techniques, and hybrid analysis methods. Additionally, we highlighted real-world challenges such as polymorphic malware, adversarial attacks, and model explainability—issues echoed in studies like Gupta et al. (2021) and the Kaspersky whitepaper.

What sets our framework apart is its practical applicability—it balances high accuracy (up to 99%) with lower false-positive rates and uses optimized models that require less manual feature engineering. Even simple CNN architectures proved to be both effective and efficient, making them feasible for real-world deployment, including on limited-resource environments like IoT and embedded systems.

However, important challenges remain. Deep learning models still require substantial computational power and remain vulnerable to adversarial manipulation. Moreover, the "black-box" nature of these models limits trust and transparency in critical applications.

To move forward, future research must focus on:

- Developing Explainable AI (XAI) to enhance model transparency.
- Creating adversarially robust models capable of withstanding evasion techniques.
- Incorporating transfer learning to improve generalization across platforms and datasets.
- Expanding detection frameworks to IoT environments where lightweight and adaptive models are crucial.
- Exploring ensemble and hybrid approaches for even greater detection reliability.

In essence, this research contributes meaningfully to the ongoing shift toward intelligent, adaptive malware detection systems that are not only accurate but also more resilient, transparent, and scalable for real-world cybersecurity needs.

6. Conclusion

As cyber threats grow more sophisticated, traditional signature-based detection methods are no longer sufficient—especially against zero-day and polymorphic malware. This research highlights how machine learning, particularly behavior-based and dynamic analysis, offers a more adaptive and intelligent approach to malware detection.

By analyzing real-time system behavior such as CPU usage and API calls, our ML-driven framework can detect threats that static methods often miss. The results show that dynamic detection models not only improve accuracy but also offer better resilience in real-world scenarios.

While dynamic analysis introduces added complexity, its long-term benefits justify the trade-off. This study emphasizes the need for scalable, explainable, and behavior-aware detection systems. Looking forward, technologies like continual learning, federated learning, and edge-based detection can further enhance real-time threat identification across diverse environments, including IoT devices.

Embedding transparency through Explainable AI (XAI) will also be critical to building trust in automated cybersecurity solutions. Ultimately, the future of malware detection lies in developing intelligent, autonomous systems that can not only detect but also anticipate and neutralize threats proactively.

7. REFERENCES

1.https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8681127
2.https://ieeexplore.ieee.org/ielx7/6287639/10005208/10068497.pdf?tp=&arnumber=10068497&isnumber=10005208&ref=aHR0cHM6Ly9pZWVleHBsb3JlLmllZWUub3JnL2RvY3VtZW50LzEwMDY4NDk3
3.https://mdpi-res.com/d_attachment/symmetry/symmetry-14-02304/article_deploy/symmetry-14-02304.pdf?version=1667458531
4.https://mdpi-res.com/d_attachment/symmetry/symmetry-15-00123/article_deploy/symmetry-15-00123-v2.pdf?version=1673578142
5.https://www.researchgate.net/publication/351030106_Malware_Detection_Issues_and_Challenges/fulltext/609af6c9a6fdccaebd251eb0/Malware-Detection-Issues-and-Challenges.pdf?origin=publication_detail&_tp=eyJjb250ZXh0Ijp7ImZpcnN0UGFnZSI6InB1YmxpY2F0aW9uIiwicGFnZSI6InB1YmxpY2F0aW9uRG93bmxvYWQiLCJwcmV2aW91c1BhZ2UiOiJwdWJsaWNhdGlvbiJ9fQ
6.https://mdpi-res.com/d_attachment/electronics/electronics-14-00167/article_deploy/electronics-14-00167.pdf?version=1735889475
7.https://www.ijstr.org/final-print/jan2020/Detection-Of-Malware-Using-Deep-Learning-Techniques.pdf
8.https://media.kaspersky.com/en/enterprise-security/Kaspersky-Lab-Whitepaper-Machine-Learning.pdf
9. https://arxiv.org/pdf/2407.19153
10.https://www.researchgate.net/publication/360962137_Deep_Learning_based_Cross_Architecture_Internet_of_Things_malware_Detection_and_Classification
11. https://www.mdpi.com/2073-8994/15/1/123
12.https://onlinelibrary.wiley.com/doi/abs/10.1155/2022/2959222
13.https://ieeexplore.ieee.org/abstract/document/9936621/
14.https://www.sciencedirect.com/science/article/pii/S1877050921007481
15.https://link.springer.com/chapter/10.1007/978-3-030-04780-1_28
16. https://arxiv.org/pdf/2303.16004
17. https://arxiv.org/pdf/1710.08189
18.https://www.researchgate.net/profile/Hoda-El-Merabet/publication/330826706_A_Survey_of_Malware_Detection_Techniques_based_on_Machine_Learning/links/5c61a25f92851c48a9cd34bc/A-Survey-of-Malware-Detection-Techniques-based-on-Machine-Learning.pdf?_tp=eyJjb250ZXh0Ijp7ImZpcnN0UGFnZSI6InB1YmxpY2F0aW9uIiwicGFnZSI6InB1YmxpY2F0aW9uIn19
19. https://arxiv.org/pdf/2006.09271
20.https://www.ijraset.com/best-journal/malware-detection-using-machine-learning
21. https://www.mdpi.com/2073-8994/14/11/2304
22.https://www.researchsquare.com/article/rs-4219382/latest