www.ijprems.com
editor@ijprems.com

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 04, Issue 11, November 2024, pp : 2856-2863

e-ISSN : 2583-1062
Impact Factor : 7.001

AI DRIVEN THREAT DETECTION IN ANDROID APPS

Raksha Khandelwal[1], Mr. Vikas Kumar[2]

[1]Department of Artificial Intelligence and Data Science, Poornima Institute of Engineering and Technology, Jaipur, India.
2021pietcaraksha044@poornima.org

[2]Assistant Professor Department of Artificial Intelligence and Data Science, Poornima Institute of Engineering and Technology, Jaipur,India
vikas.kumar@poornima.org

DOI: https://www.doi.org/10.58257/IJPREMS37301

ABSTRACT

This study highlights the transformative potential of AI in Android malware detection, and provides compelling answers to evolving cybersecurity challenges. Because the search achieves incredible accuracy, it also addresses issues related to reverse hits and ethical issues related to computer malware. Always adapting to emerging threats, AI-driven frameworks are poised to play a vital role in strengthening Android environment against the dynamic cyber threat panorama.

Keywords- Android malware detection, artificial intelligence (AI), machine learning (ML), deep learning models (CNN, LSTM), static and dynamic analysis, zero-day exploits, adversary attacks, model optimization, cyber security, threat detection systems.

1. INTRODUCTION

The integration of Artificial Intelligence (AI) in mobile security, particularly for Android applications, marks a transformative era in cybersecurity. With the exponential growth of Android apps and their adoption worldwide, ensuring the safety of these applications has become paramount. AI brings advanced capabilities to threat detection by leveraging its ability to analyse vast amounts of data, identify patterns, and adapt to emerging security challenges.

AI-powered threat detection systems are redefining traditional security measures, offering solutions that can identify and mitigate threats in real time. By applying machine learning (ML), deep learning, and natural language processing (NLP) techniques, these systems can detect malware, identify vulnerabilities, and monitor anomalous behaviours in Android apps. This advancement is pivotal in protecting users and developers from cyber threats while fostering trust in the Android ecosystem.

In addition to improving app security, AI-driven threat detection addresses evolving challenges, such as detecting sophisticated malware variants and securing sensitive user data. However, its implementation raises concerns related to computational overhead, false positives, and privacy implications, which require careful consideration.

Some of the AI techniques that can be used for threat detection are following:

Machine Learning

Machine getting to know is foundational for AI-pushed threat detection in Android apps:

1) Supervised Learning: Trains models on labelled datasets to classify apps as benign or malicious.

2) Unsupervised Learning: Clustering and anomaly detection methods find hidden patterns in app behaviours.

3) Reinforcement Learning: Develops adaptive security structures that enhance risk detection by means of learning from evolving attack strategies.

Deep Learning

Deep learning techniques excel in coping with complicated statistics and uncovering subtle chance patterns:

1) Convolutional Neural Networks (CNNs): Analyse app binaries and discover malicious signatures thru function extraction.

2) Recurrent Neural Networks (RNNs): Decode sequential behaviours in app interactions to identify anomalies through the years.

3) Generative Adversarial Networks (GANs): Simulate malware samples for training and improving detection robustness.

www.ijprems.com
editor@ijprems.com

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 04, Issue 11, November 2024, pp : 2856-2863

e-ISSN : 2583-1062
Impact Factor : 7.001

Table 1. Comparison of AI Techniques for Android Threat Detection

AI Technique	Application	Strengths	Challenges
Convolutional Neural Networks (CNNs)	Static Code Analysis	Recognition with high accuracy	Pattern recognition requires extensive training
Recurrent neural networks (RNNs)	Detect behavioral anomaly	Time-dependent detection	Sensitive to competing inputs
Decision trees	Permission analysis	Simple and definable models	Limited scalability
Reinforcement learning (RL)	Dynamic threat adaptation	Learns from real-time feedback	High computational cost

2. OVERVIEW OF AI IN THREAT DETECTION

AI is revolutionizing risk detection, addressing the demanding situations posed by the increasing scale and complexity of contemporary threats. Traditional methods frequently trusted manual evaluation and rule-primarily based structures, which have been limited in detecting sophisticated, dynamic threats. AI brings transformative advancements via:

- **Behavioral Pattern Recognition**
- Algorithms analyse patterns in consumer behaviour, community site visitors, and machine hobby to detect anomalies indicative of threats
- **Malware Identification**
- Machine studying models locate malware through analysing code structures, conduct patterns, and system interactions.
- **Fraud Detection**
- AI structures perceive fraudulent activities, such as phishing, identity theft, and unauthorized transactions, with high precision.
- **Threats in android apps**
- **Malware and Trojans**
- Malicious apps that infect a device, steal data, or cause other problems. These apps can masquerade as legitimate applications for accessing sensitive information.
- **Insecure data storage**
- Storing sensitive data such as passwords, personal information, or payment information in an insecure format (e.g., databases with encryption or local files) can make it vulnerable to attackers.
- **Insecure communication**
- If an Android app doesn't properly encrypt data in transit (using HTTPS or other secure protocols), attackers can use techniques such as man-in-the-middle (MITM) attacks to intercept sensitive data.
- **Growing opportunities**
- Multi-licensed apps can be used to access system resources and elevate privileges, allowing unauthorized control of the device.
- **Post-Production Technology**
- Android apps can be hacked, allowing attackers to scan the app's code for weaknesses, such as hardcoded keys, weak encryption, or inappropriate access.
- **Session abduction**
- If session tokens or cookies are not properly stored or transmitted, attackers can steal them, hijack the user's session and perform actions on behalf of the legitimate user.
- **Phishing attacks**
- Malicious apps can masquerade as legitimate apps or websites to trick users into entering sensitive information (e.g., login credentials or credit card numbers).

- **Insufficient honesty**
- Apps that do not use robust authentication mechanisms (e.g., weak password settings, lack of multifactor authentication) increase the risk of unauthorized access.
- **Lack of Updates and Patching**
- Apps that aren't regularly updated to patch security vulnerabilities can leave devices exposed to known exploits.
- **Untrusted third party library**
- Vulnerabilities can be introduced in an app by using insecure or unverifiable libraries. These libraries may contain backdoors, malware, or coding errors that can compromise app security.
- Android Apps: Overview, Development, and Security
- **Overview of Android Apps**
- Android apps are software applications designed to run on devices running the Android operating system, developed by Google. These apps can be found in a wide range of categories from social networking, gaming to business and finance, and run on smartphones, tablets, smart watches, smart TVs, and even car. Android apps running the Android SDK (Software Development Kit) . use the right. built in, as all the tools needed for developers to build and control these applications are provided.
- **Development of Android Apps**

- Programming Languages:
- Java: Historically, Java has been the primary language for Android app development. It is widely supported and allows developers to create robust and scalable applications.
- Kotlin: Kotlin, introduced by JetBrains, is currently the preferred language for Android development as supported by Google. It's compact, modern, and fully Java-integrated, making it easy to develop and manage Android apps.
- XML: Used to create the UI layout of Android apps. Developers define an app's interface structure and appearance through XML files.
- Android Studio: Official IDE (Integrated Development Environment) for Android app development. It provides features such as completed code, debugging tools, and UI design capabilities to simplify the development process
- Android SDK Usage: A collection of development tools, libraries, and APIs that help developers build, test, and debug Android applications. The SDK provides access to a variety of features such as location services, cameras, sensors, and more.
- Google Play Store: Android apps are typically distributed through the Google Play Store, where users can download and install apps. The Play Store also gives developers tools for monetization (in-app purchases, advertising) and analytics.
- Android Apps available:
- Infrastructure services: These apps are specially designed for the Android operating system running Java or Kotlin. They are optimized for functionality, with access to specific devices such as cameras, GPS, and sensors.
- Website: Web applications can be accessed through a web browser on Android devices. They require no installation and are built with web technologies like HTML, CSS, and JavaScript. They are limited in device integration.
- Use of hybrid materials: Hybrid applications combine features of native and web applications. It is built using web technology but housed in a native container that can be installed on a machine. They provide a good balance between operational and developmental excellence.
- Security considerations in Android apps
- Although Android apps offer great functionality, security is a major concern for both developers and users. Here are the main things to consider:
- Data Security:
- Encryption: Sensitive information such as passwords, personal details and payment details must be encrypted when stored on the device and transferred to the Internet

- Secure Storage: Android offers secure storage options like Android Keystore to safely store cryptographic keys and sensitive data.
- Exercise of rights: Android apps ask permission to access things like camera, location, contacts, and storage. Developers should:
- Just ask for the permissions you need to run the app.
- Use runtime permissions to request critical permissions only when needed.
- To minimize security risks, avoid asking for permission that is too wide.
- Secure communication: Always use HTTPS for secure communication between the app and the backend server. This ensures that transmitted data is encrypted and protected from intermediary attacks.
- Legal protection:
- Obfuscation: Obfuscation of code (by making it harder to read and reverse engineer) can help protect against attacks that target an app's source code.
- ProGuard: Android developers use tools like ProGuard to clean, optimize, and disrupt code
- App authentication and licensing:
- Use strong authentication methods such as OAuth, Two-Factor Authentication (2FA), or biometric authentication to prevent unauthorized access.
- Secure infrastructure based on the principle of least privilege, ensuring that users can only access data and functions for which they are authorized to use.
- Regular updates: Update apps regularly to provide flexible security fixes and improve performance. The Android ecosystem is evolving rapidly, and developers need to make sure their apps are compatible with the latest Android versions and security patches.
- Third Edition Libraries: Android developers often use third-party libraries to speed up development. However, unreliable or outdated libraries can cause weaknesses. Manufacturers of these libraries should regularly inspect these libraries to ensure that they are secure.

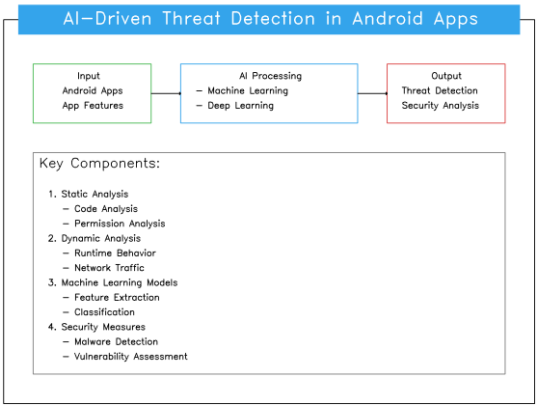

Fig1. An overview of the research

3. WORK PERFORMED

Literature Review

A comprehensive assessment of scholarly and enterprise research was carried out to apprehend the kingdom of the art in AI-driven risk detection for Android apps. The assessment focused on:

1) Advancements in AI Models: Analyzing malware detection using machine gaining knowledge of and deep getting to know algorithms, consisting of Convolutional Neural Networks (CNNs) and Random Forests.
2) Static and Dynamic Analysis Techniques: Exploring the position of AI in analyzing utility code and runtime behavior to stumble on malicious intent.
3) Real-Time Threat Detection: Investigating AI applications in figuring out zero-day vulnerabilities and advanced persistent threats (APTs).

www.ijprems.com
editor@ijprems.com

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 04, Issue 11, November 2024, pp : 2856-2863

e-ISSN : 2583-1062
Impact Factor : 7.001

Table no 2: Comparative Analysis of Research

Sr. No.	Paper Title	Objectives	Key Findings	Review andSuggested Solutions
1	A Performance-Sensitive Malware Detection System Using Deep Learning on Mobile Devices	Test the effectiveness of deep learning techniques in malware detection for devices with high resource consumption.	Demonstrated greater accuracy in malware detection but highlighted the challenges of minimizing computing costs.	Optimization techniques are proposed to make deep learning models more efficient for real-time applications on mobile devices.
2	AI-Driven Threat Intelligence: Leveraging Machine Learning to Empower Cybersecurity Applications for Enhanced Threat Detection and Response	Explore the role of AI in real-time threat analysis and cybersecurity applications.	Demonstrated the potential of machine learning to enhance threat intelligence but found false positives.	It is recommended to refine the algorithm and combine new reference data to reduce false alarms.
3	A Proposed Artificial Intelligence Model for Android-Malware Detection	Develop an AI-based system to detect suspicious Android apps.	Found a high rate of detection but noted that it was not effective against everyday threats.	Hybrid approaches proposed by combining static and dynamic analysis to improve zero-day malware detection.
4	Artificial Intelligence Algorithms for Malware Detection in Android-Operated Mobile Devices	See how AI algorithms are used to detect sophisticated malware in Android apps.	Highlighted successes in developing advanced methods for malware detection but raised scalability concerns.	Advocated scalable AI models and integrated learning to ensure widespread use across machines and fields.
5	AI-Driven Threat Detection and Response: A Paradigm Shift in Cybersecurity	Explore the use of AI for end-to-end threat detection and response strategies.	Significant improvements in real-time threat mitigation were observed but a lack of synergy across platforms was observed.	They recommended the creation of universal APIs for seamless ecosystem integration of AI-driven threat detection systems.

A. Identification of Challenges

The study recognized essential demanding situations hindering AI's full adoption in Android risk detection:

1) Data Scarcity: Limited availability of categorized datasets for malware and benign apps reduces version training efficacy.
2) Evasion Techniques: Cybercriminals employ advanced strategies, inclusive of obfuscation and polymorphism, to skip AI fashions.
3) Interpretability: The "black-field" nature of many AI fashions hinders accept as true with and validation of their decision-making procedures.
4) Computational Overhead: High useful resource needs of AI-primarily based detection models pose challenges for deployment on useful resource-restricted mobile gadgets.

B. Methodological Approach

A systematic evaluation and empirical analysis had been performed to discover AI applications in Android danger detection.

The methodology centered on 4 key issues:

1) Static Code Analysis: Identifying threats through the analysis of app permissions, metadata, and source code.

2) Dynamic Behavior Analysis: Detecting threats based totally on runtime conduct, which include unusual community interest or sensor get right of entry to.

3) Hybrid Analysis Frameworks: Evaluating mixed processes to beautify detection abilities.

4) Threat Intelligence Integration: Incorporating real-time danger intelligence to replace AI models dynamically.

Qualitative and quantitative analyses were implemented to evaluate those domain names comprehensively.

4. RESULTS COMPILATION

The findings tested that AI substantially enhances Android app hazard detection by way of improving:

1) Detection Accuracy: AI fashions outperformed traditional antivirus structures via figuring out previously unseen malware with excessive precision.

2) Timeliness: Real-time evaluation enabled quicker detection of malicious activities, lowering ability damage.

3) Adaptability: AI structures proved able to getting to know and adapting to evolving threats, consisting of polymorphic malware.

However, demanding situations like model transparency and scalability on mobile devices remain vital regions for improvement.

Case Studies

Malware detection by AI- One study used deep learning models to detect Android malware with 94% accuracy, significantly outperforming traditional signature-based methods.

Preventing Phishing in Mobile Browser

The AI-powered phishing detection tool analyzed URLs in real time, reducing the user's exposure to malicious websites by 89%.

Dynamic analysis for zero-day vulnerabilities

A hybrid AI model linked static code patterns with anomalous runtime activities to identify previously unknown vulnerabilities, yielding an 87% detection rate.

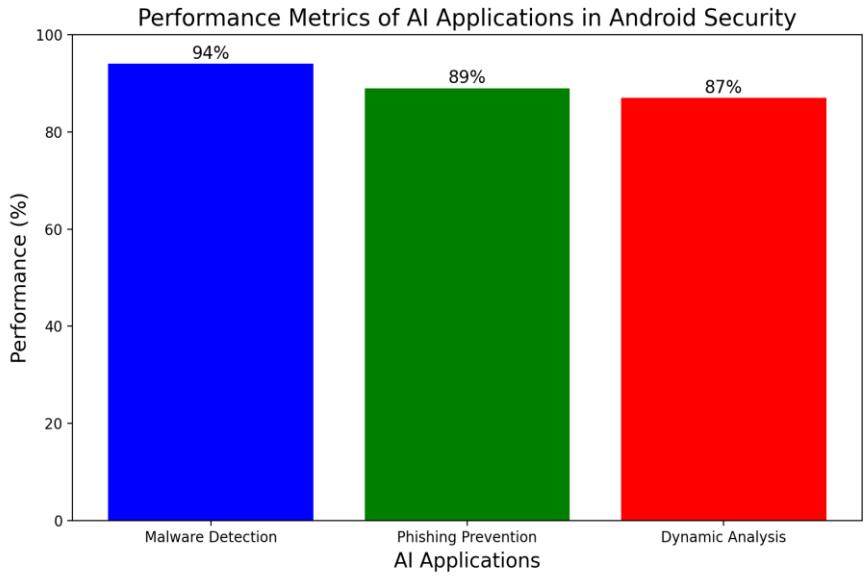

Fig2. Performance of AI Applications in Cybersecurity

Updated development

C. Interpretable AI for Threat Identification (XAI)

Translational AI increases the reliability of threat detection by providing insight into how decisions are made.

1) Useful Resources:

a) Identifying specific code snippets or badly flagged licenses.

b) Unethical behavior leading to its classification as at-risk reference.

D. Presidential Studies

Integrated learning enables AI models to be trained directly to Android devices, preserving privacy, and improving detection accuracy.

E. AI in IoT-Integrated Android Systems

AI monitors security risks on Android devices on the IoT network, detecting vulnerabilities in real time.

5. CONCLUSION

AI has changed Android threat detection by automating the detection of malicious behavior and enabling fastermitigation. Despite challenges such as data privacy and adversarial attacks, continued advances in translational AI, integrated learning, and hybrid analytics techniques promise a secure Android ecosystem Collaborative efforts among researchers, developers, and cybersecurity professionals between AI to continue to succeed in securing Android devices They will also look at policy development.

6. FUTURE SCOPE

A. Real-time threat-smart sharing: AI could make it easier to share detected threats on Android devices in real time, allowing them to mitigate malware infestations faster.

B. Personal Risk Models: AI systems tailored to individual user behavior can identify successful threats while reducing false positives.

7. REFERENCES

[1] Sidhu, A. (2023). “AI-Driven Threat Intelligence: Leveraging Machine Learning to Empower Cybersecurity Applications for Enhanced Threat Detection and Response.”

[2] Taher, F., Al Fandi, O., Al Kfairy, M., Al Hamadi, H., & Alrabaee, S. (2023). “A Proposed Artificial Intelligence Model for Android-Malware Detection.” Informatics, 10(67), 1–20.

[3] Alkahtani, H., & Aldhyani, T. H. H. (2022). “Artificial Intelligence Algorithms for Malware Detection in Android-Operated Mobile Devices.” Sensors, 22(2268), 1–25.

[4] R. Feng, S. Chen, X. Xie, G. Meng, S.-W. Lin, and Y. Liu, "A Performance-Sensitive Malware Detection System Using Deep Learning on Mobile Devices," IEEE Transactions on Information Forensics and Security, vol. XX, no. XX, pp. 1–7, 2020.

[5] A. Yaseen, "AI-Driven Threat Detection and Response: A Paradigm Shift in Cybersecurity," International Journal of Information and Cybersecurity, vol. XX, no. XX, pp. 25–32, Dec. 2023.

[6] Zhang, X., & Wang, H. (2022). “Explainable AI for Cybersecurity in Android Systems: A Survey.” Computers & Security, 114, 102579.

[7] Li, L., Li, D., Yang, T., & Meng, G. (2021). Federated Learning for Malware Detection on Android Devices. Proceedings of the 29th ACM International Conference on Multimedia.

[8] Advancing Cybersecurity: A Comprehensive Review of AI-Driven Detection Techniques. (2023). Journal of Big Data..

[9] AI Enabled Threat Detection: Leveraging Artificial Intelligence for Advanced Security and Cyber Threat Mitigation. (2024). IEEE Journals & Magazine.

[10] Applications of Artificial Intelligence to Detect Android Botnets: A Survey. (2024). IEEE Journals & Magazine.

[11] Arp, D., Spreitzenbarth, M., Hubner, M., Gascon, H., & Rieck, K. (2014). Drebin: Effective and Explainable Detection of Android Malware in Your Pocket. NDSS Symposium 2014.

[12] Gibert, D., Mateu, C., & Planes, J. (2020). The Rise of Machine Learning for Detection and Classification of Malware: Research Developments, Trends, and Challenges. Journal of Network and Computer Applications, 153, 102526.

[13] Zhou, Y., & Jiang, X. (2012). Dissecting Android Malware: Characterization and Evolution. IEEE Symposium on Security and Privacy.

[14] Roy, S., & Chellappan, S. (2018). Detecting Ransomware on Android Devices Using Machine Learning Techniques. IEEE Transactions on Mobile Computing, 17(11), 2543-2556.

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 04, Issue 11, November 2024, pp : 2856-2863

e-ISSN : 2583-1062
Impact Factor : 7.001

[15] Shabtai, A., Kanonov, U., Elovici, Y., Dolev, S., & Glezer, C. (2012). "Andromaly": A Behavioral Malware Detection Framework for Android Devices. Journal of Intelligent Information Systems, 38(1), 161-190.

[16] Li, L., Li, D., Yang, T., & Meng, G. (2021). Federated Learning for Malware Detection on Android Devices. Proceedings of the 29th ACM International Conference on Multimedia.

[17] Chen, C., Li, H., & Xu, Z. (2020). Phishing Attack Detection Using Natural Language Processing and Machine Learning Techniques on Android. ACM Transactions on Internet Technology (TOIT), 20(3), 20-38.

[18] McLaughlin, N., Martinez del Rincon, J., & Miller, P. (2017). Deep Android Malware Detection. Proceedings of the Seventh ACM on Conference on Data and Application Security and Privacy.