UDC 004.056.53

THE ROLE OF ARTIFICIAL INTELLIGENCE IN MALICIOUS CODE ANALYSIS

K.S. Artyklyyev, I.G. Tagangylyjov

Oguz han Engineering and Technology University of Turkmenistan, Ashgabat, Turkmenistan,
kartyklyyew@gmail.com

Abstract: this paper examines the role of modern artificial intelligence (AI) methods in the detection and analysis of malicious software. The main approaches are reviewed-static, dynamic, and hybrid analysis based on machine learning (ML) and deep learning (DL) algorithms-together with the practical problems that limit their deployment: code obfuscation, adversarial attacks on models, dataset scarcity and drift, limited interpretability, and detection of zero-day threats. It is shown that combining classical signature-based methods with AI-driven models improves detection accuracy, reduces analyst workload, and shortens response time to emerging threats. Directions for further research, including explainable AI and adversarially robust models, are outlined.

Keywords: artificial intelligence, machine learning, malware analysis, cybersecurity, intrusion detection, deep learning, adversarial robustness.

I. INTRODUCTION

The growing volume and sophistication of malicious software (malware) samples continually erodes the effectiveness of traditional signature-based detection. Polymorphism, metamorphism, packing, and obfuscation allow attackers to generate functionally identical payloads with different byte-level representations, letting them slip past static signature databases with minimal effort. At the same time, the sheer scale of newly observed samples-commonly cited at several hundred thousand per day across major vendors-makes manual analyst review of every file impractical [1-6].

Under these conditions, artificial intelligence, and in particular machine learning and deep learning, has become a central tool for automated malware analysis. Rather than requiring an exact match to a known signature, AI-based systems learn statistical and structural regularities that separate malicious from benign software, which allows them to generalize to previously unseen variants and, in some cases, to entirely new families.

II. STATIC ANALYSIS WITH AI

Static analysis examines a file without executing it, extracting features such as byte and opcode sequences, imported API calls, string constants, entropy measures, and the structure of executable headers (PE, ELF). These features are commonly represented as fixed-length vectors, n-gram frequency tables, or, in more recent work, as sequences fed directly into deep neural networks.

Classical machine learning classifiers-decision trees, random forests, support vector machines, and gradient boosting-remain popular for static detection because of their relatively low computational cost, fast inference, and reasonable interpretability compared to deep architectures. Convolutional neural networks (CNNs) have also been applied by treating the raw binary as a grayscale image, allowing the network to learn local structural patterns directly from byte layout without manual feature engineering.

The principal weakness of static analysis is its vulnerability to obfuscation: packing, encryption of the payload, and code virtualization can substantially alter the extracted features while leaving the underlying malicious behavior unchanged, which reduces classifier accuracy unless combined with unpacking or emulation steps.

III. DYNAMIC AND BEHAVIORAL ANALYSIS

Dynamic analysis executes a sample in an isolated environment, such as a sandbox or virtual machine, and records its runtime behavior: system and API calls, file system and registry modifications, process creation, and network traffic. Because this behavior reflects what a program actually does rather than how it is encoded, dynamic features are considerably more resistant to obfuscation than static ones.

The resulting event sequences are naturally modeled as time series, which motivates the use of recurrent architectures such as long short-term memory (LSTM) and gated recurrent unit (GRU) networks, as well as, more recently, transformer-based models capable of capturing long-range dependencies between events. Such models can distinguish behavioral patterns characteristic of specific malware families-for example, the mass file-encryption activity typical of ransomware, the credential-harvesting behavior of remote access trojans, or the persistence mechanisms used by rootkits.

The main drawback of dynamic analysis is cost: sandbox execution is slower than static feature extraction, and sophisticated malware can detect a virtualized or monitored environment and suppress its malicious behavior, a technique known as sandbox evasion.

IV. HYBRID AND GRAPH-BASED APPROACHES

Hybrid approaches combine static and dynamic features to compensate for the weaknesses of each method individually. A further direction that has gained attention is the use of graph neural networks (GNNs) applied to control-flow graphs, call graphs, or data-dependency graphs extracted from a binary. Because such graphs capture the logical structure of a program rather than its literal encoding, GNN-based models tend to be more robust to obfuscation and code transformation than purely sequence-based models, while still allowing static, pre-execution analysis.

Ensemble methods that combine the outputs of several models-for instance, a static classifier, a behavioral classifier, and a graph-based classifier-are also widely used in production security products, since disagreement between independent models can itself serve as a useful signal for flagging suspicious files for human review.

V. CHALLENGES AND LIMITATIONS

Despite significant progress, several open problems limit the deployment of AI in malware analysis. First, ML and DL models are vulnerable to adversarial attacks: small, often imperceptible perturbations to a file's features can cause a misclassification, and attackers can craft malware specifically to evade a known detection model. Second, the quality and freshness of training data are persistent concerns-malware datasets become outdated quickly as new families and evasion techniques emerge, and public datasets are often imbalanced or unrepresentative of real-world traffic, a phenomenon known as concept drift. Third, the interpretability of deep learning verdicts remains low, which complicates their use in settings where an analyst must justify a decision, for example in incident response or legal proceedings. Finally, models trained primarily on one platform or malware family often generalize poorly to others, requiring continual retraining and monitoring in production.

VI. FUTURE DIRECTIONS

Several directions appear promising for addressing the above limitations. Explainable AI (XAI) techniques, such as feature attribution and attention visualization, can help analysts understand and validate model verdicts. Adversarial training and robust optimization can improve resilience against evasion attempts. Federated and continual learning approaches may allow models to be updated with new threat data without requiring full retraining or centralizing sensitive samples. Finally, closer integration between AI-based detectors and threat-intelligence platforms can help models adapt more quickly to newly disclosed vulnerabilities and campaigns.

VII. CONCLUSION

Artificial intelligence substantially extends the capabilities of malware analysis, enabling the detection of previously unknown threats and reducing the workload placed on human analysts through automation. Static, dynamic, and hybrid AI-based methods each offer complementary strengths, and their combination in ensemble systems currently represents best practice in production environments. Further progress in the field depends on improving model robustness against adversarial manipulation, developing explainable models suitable for security-critical decision-making, and maintaining up-to-date, representative training data.

REFERENCES

[1] Ucci, D., Aniello, L., Baldoni, R. Survey of machine learning techniques for malware analysis // Computers & Security. – 2019. – Vol. 81. – P. 123–147.

[2] Gibert, D., Mateu, C., Planes, J. The rise of machine learning for detection and classification of malware: Research developments, trends and challenges // Journal of Network and Computer Applications. – 2020. – Vol. 153.

[3] Raff, E., Nicholas, C. Survey of machine learning methods and challenges for Windows malware classification // arXiv preprint. – 2020.

[4] Rosenberg, I., Shabtai, A., Elovici, Y., Rokach, L. Adversarial machine learning attacks and defense methods in the cyber security domain // ACM Computing Surveys. – 2021. – Vol. 54, No. 5. – P. 1–36.

[5] Alsulami, B., Srinivasan, A., Dong, H., Mancoridis, S. Lightweight behavioral malware detection for windows platforms // Proceedings of the 12th International Conference on Malicious and Unwanted Software (MALWARE). – 2017.

[6] Wu, Y., et al. Malware classification using attention-based graph neural networks // IEEE Access. – 2021.

РОЛЬ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА В АНАЛИЗЕ ВРЕДОНОСНОГО КОДА

Артыклыев К.С., Тагангылыджов И.Г.

Огузханский инженерно-технологический университет Туркменистана, Ашхабад, Туркменистан

Аннотация: в статье рассматривается роль современных методов искусственного интеллекта (ИИ) в обнаружении и анализе вредоносного программного обеспечения. Рассмотрены основные подходы — статический, динамический и гибридный анализ на основе алгоритмов машинного (МО) и глубокого обучения (ГО), а также практические проблемы, ограничивающие их применение: обфускация кода, состязательные атаки на модели, нехватка и дрейф обучающих данных, низкая интерпретируемость и обнаружение угроз нулевого дня. Показано, что сочетание классических сигнатурных методов с моделями на основе ИИ повышает точность обнаружения, снижает нагрузку на аналитиков и сокращает время реагирования на новые угрозы. Намечены направления дальнейших исследований, включая объяснимый ИИ и устойчивые к атакам модели.

Ключевые слова: искусственный интеллект, машинное обучение, анализ вредоносного ПО, кибербезопасность, обнаружение вторжений, глубокое обучение, устойчивость к атакам.