Leveraging Spectral Representations of Control Flow Graphs for Efficient Analysis of Windows Malware

Qirui Sun
Sungkyunkwan University
Suwon, South Korea
qirui@g.skku.edu

Eldor Abdukhamidov
Sungkyunkwan University
Suwon, South Korea
abdukhamidov@skku.edu

Tamer Abuhmed†
Sungkyunkwan University
Suwon, South Korea
tamer@skku.edu

Mohammed Abuhamad†
Loyola University Chicago
Chicago, United States
mabuhamad@luc.edu

ABSTRACT

The rapid pace of malware development and the widespread use of code obfuscation, polymorphism, and morphing techniques pose a considerable challenge to detecting and analyzing malware. Today, it is difficult for antivirus applications to use traditional signature-based detection methods to detect morphing malware. Thus, the emergence of structure graph-based detection methods has become a hope to solve this challenge. In this work, we propose a method for detecting malware using graphs' spectral heat and wave signatures, which are efficient and size- and permutation-invariant. We extracted 250 and 1,000 heat and wave representations, and we trained and tested heat and wave representations on eight machine learning classifiers. We used a dataset of 37,537 unpacked Windows malware executables and extracted the control flow graph (CFG) of each windows malware to obtain the spectral representations. Our experimental results showed that by using heat and wave spectral graph theory, the best malware analysis accuracy reached 95.9%.

CCS CONCEPTS

• **Security and privacy → Malware and its mitigation**; **Intrusion/anomaly detection and malware mitigation**.

KEYWORDS

Malware Detection, Spectral Representation, Windows Executable Binaries, Size-invariant Representations

ACM Reference Format:
Qirui Sun, Eldor Abdukhamidov, Tamer Abuhmed†, and Mohammed Abuhamad†. 2022. Leveraging Spectral Representations of Control Flow Graphs for Efficient Analysis of Windows Malware. In *Proceedings of the 2022 ACM Asia Conference on Computer and Communications Security (ASIA CCS '22), May 30-June 3, 2022, Nagasaki, Japan.* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/3488932.3527294

† Corresponding Author.

ASIA CCS '22, May 30-June 3, 2022, Nagasaki, Japan

ACM ISBN 978-1-4503-9140-5/22/05.
https://doi.org/10.1145/3488932.3527294

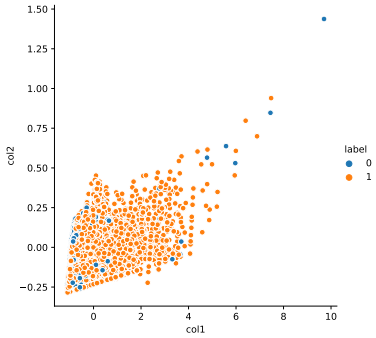
1 INTRODUCTION

In recent years, malware variants have shown a rapid development trend, and malware designers are using new techniques such as packing, deformation, polymorphism, and code obfuscation to avoid anti-malware software detection. Therefore, it becomes easier to create new malicious software using these novel technologies and tools, while traditional detection methods, such as standard signature-based detection, have been facing many challenges to keep up-to-date with such an increasing trend. Innovative and novel techniques that are based on pattern-mining and execution/memory forensics have become a new hot topic.

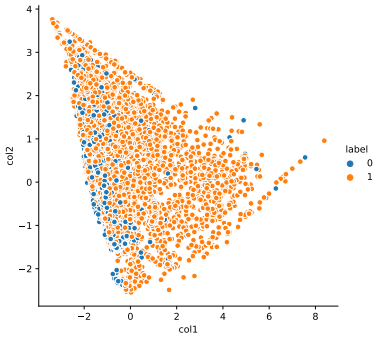
Facilitating the analysis from both static and dynamic artifacts, machine learning ML-based malware detection methods have become an integral part of solutions in industrial and academic exploration [1, 3, 6]. Since machine learning algorithms can explore in-depth relationships between traits and the output, they fully exploit information about malicious code to determine their behavior. Consequently, machine learning-based malicious code detection tends to exhibit high accuracy rates and discovers/automates the analysis of unknown malicious code. Furthermore, the breakthrough progress of deep learning technology in computer vision, speech recognition, natural language processing, and other domains has provided a new perspective for malware detection research in recent years [4]. Deep learning models can automatically learn feature representations from the binary's raw, unstructured bytes.

From another direction, many graph-based research methods have emerged in recent years, contributing to code analysis. Graph structures, such as abstract syntax trees, control-flow graphs (CFGs), and data-flow graphs, can be extracted and studied to obtain meaningful features. One challenge, however, is to maintain both expressiveness and efficiency when representing graphs, especially when analyzing graphs with different sizes. NetLSD [7] offers efficient and size- and permutation-invariant graph representations.

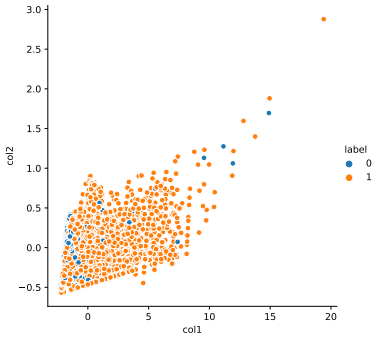
Since machine learning-based methods are highly dependent on extracted features and classifiers, it is crucial to explore multiple algorithms using artifacts generated by several techniques to achieve high detection rates and low false-positive rates. In this work, we propose a novel approach that utilizes spectral graph theory to capture heat and wave features and apply them to eight classifiers to improve malware detection accuracy.



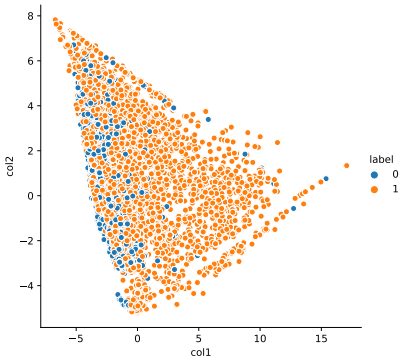
(a) 250-dimensional heat spectral representations



(b) 250-dimensional wave spectral representations



(c) 1000-dimensional heat spectral representations



(d) 1000-dimensional wave spectral representations

Figure 1: PCA analysis of benign and malware samples on heat and wave kernel representations

Contributions. The contributions of this work are as follows: ❶ We investigate the performance of eight machine learning-based malware detection methods using a dataset of 37,537 Windows executable samples. ❷ We use spectral representations of control flow graphs to extract 250- and 1000-dimensional heat and wave representations, apply them to machine learning classifiers, and achieve a detection accuracy of 95.9%.

2 METHODS

This section describes the dataset, the executable extracted artifacts and their spectral representations, the explored machine learning methods, and the experimental settings and evaluation metrics.

Dataset. In this work, we use the malware dataset provided by Aghakhani *et al.* [2]. The dataset consists of the dataset from the EMBER, and a commercial vendor, and it contains an overall 37,537 samples, including 12,472 unpacked benign programs and 25,065 unpacked malicious executables.

Artifacts and Feature Representations. This work adopts a static analysis approach to analyze executable binaries. Specifically, we study the CFG for traits and behavior of maliciousness, which is effective and accurate from malware analysis. We use NetLSD to extract heat and wave representations of CFGs. To this end, we adopted the following steps.

First of all, using radare2's [5] Python API – r2pipe, we extracted the CFGs for all Windows executable binaries. Then, we applied CFGs to NetLSD to generate 250- and 1,000-dimensional heat and wave graph spectral representation. NetLSD generates compact graph signatures based on the Laplacian's heat or wave kernel, which inherit the Laplacian spectrum's formal features. The heat kernel is a family of low-pass filters that captures low-frequency information in the graph at every scale. Wave kernel maintains symmetries and structures on the spectrum via band-pass filters.

In Figure 1, we use Principal Component Analysis (PCA) to visualize the 250- and 1000-dimensional heat and wave representations. PCA is used for dimensionality reduction, where each data point is projected onto only the first few principal components to generate lower-dimensional data while preserving as much variation as possible. As a result, col1 and col2 represent the features after the dimension reduction, label 0 is benign, and label 1 is malware. As you can see in Figure 1, the majority of samples are malware. PCA analysis shows that benign samples share a common similarity as they are located near each other.

Machine Learning Methods. We applied several machine learning models for windows malware detection. The following eight machine learning algorithms are applied: ❶ Support Vector Machine (SVM), ❷ Decision Tree (DT), ❸ Logistic Regression (LR),

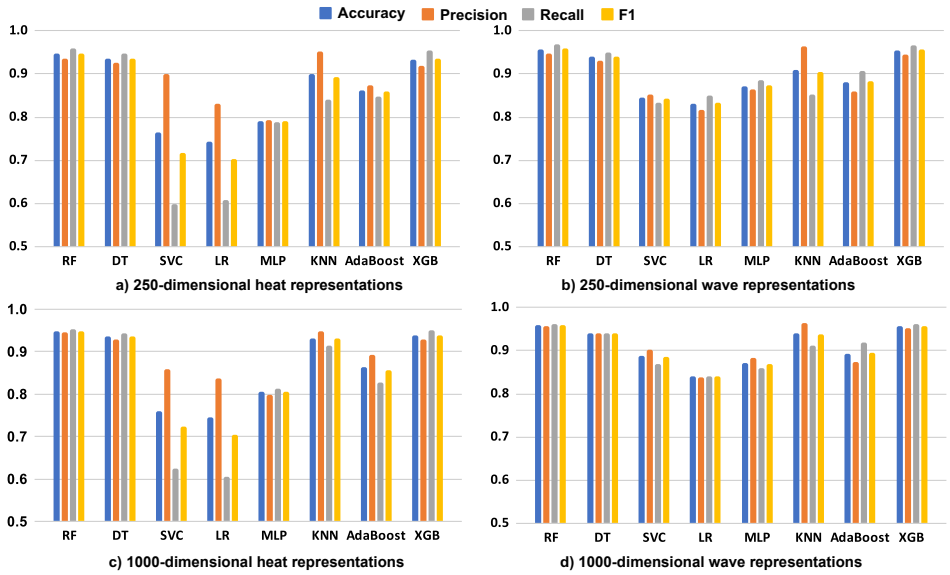


Figure 2: Results of performance metrics on eight different machine learning classifiers

❹ Random Forest (RF), ❺ K-Nearest Neighbors (KNN), ❻ artificial neural network (ANN), ❼ Adaptive Boosting (AdaBoost) and ❽XGBoost(XGB) in our experiments.

Experiment Settings. To handle class imbalance, we applied Synthetic Minority Oversampling Technique (SMOTE) to oversample the minority class. Then, we randomly split the dataset into training and testing datasets (70% and 30% respectively) before training the ML models. We repeated the experiment ten times and reported the results using accuracy, precision, recall, and F1 score.

3 PRELIMINARY RESULTS

In the experiments, we apply the 250- and 1000-dimensional heat and wave features to build classifiers and obtain the final accuracy, precision, recall, and F1 score. Each experiment is repeated ten times, with the dataset being randomly split each time to allow for impartial model evaluation, and then we report the average results as shown in Figure 2. Figure 2 (a) and (c) shows that the results of the 250 heat features experiment are similar to those of the 1000 heat features experiment. RF, DT, and XGB obtained high accuracy, precision, recall, and F1 scores, and almost all evaluation scores were in the range of 93% to 95%. On the contrary, the performance of SVC and LR is not as good, although their precision scores are high, their accuracy, recall, and F1 scores are very low. Figure 2 (b) and (d), shows that the wave features have better performance and the evaluation metrics of all eight classifiers have improved around 1% to 6% (compared to results obtained from heat representations). RF, DT, and XGB are still the best among the eight classifiers. Although the difference is insignificant, the 1000 wave representations have 0.1% to 0.2% improvement over the 250 wave representations, indicating huge possible efficiency gains, working with large datasets or inference throughput. We summarize the following experimental results: ❶ RF, DT, and XGB are the three best performing classifiers in all experiments. ❷ Wave representations perform better than heat representations. ❸ Using higher dimensional features (1000 vs. 250) does not significantly help machine learning classifiers.

4 CONCLUSION

In this paper, we propose a novel idea to detect windows malware by extracting heat and wave features through the NetLSD method using spectral graph theory. From the experimental results, we can conclude that heat and wave features are helpful for malware detection. Wave spectral representations have better performance than heat representations when using the same dimensionality. We plan to investigate the changes in spectral representations for malware variants and how perturbations impact these representations.

Acknowledgments. This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (No. 2021R1A2C1011198).

REFERENCES

[1] Ahmed Abusnaina, Mohammed Abuhamad, Hisham Alasmary, Afsah Anwar, Rhongho Jang, Saeed Salem, Daehun Nyang, and David Mohaisen. 2021. DL-FHMC: Deep Learning-based Fine-grained Hierarchical Learning Approach for Robust Malware Classification. *IEEE Transactions on Dependable and Secure Computing* (2021).

[2] Hojjat Aghakhani, Fabio Gritti, Francesco Mecca, Martina Lindorfer, Stefano Ortolani, Davide Balzarotti, Giovanni Vigna, and Christopher Kruegel. 2020. When Malware is Packin'Heat; Limits of Machine Learning Classifiers Based on Static Analysis Features. *Network and Distributed Systems Security Symposium 2020.*

[3] Hisham Alasmary, Ahmed Abusnaina, Rhongho Jang, Mohammed Abuhamad, Afsah Anwar, DaeHun Nyang, and David Mohaisen. 2020. Soteria: Detecting adversarial examples in control flow graph-based malware classifiers. In *2020 IEEE 40th International Conference on Distributed Computing Systems (ICDCS).* 888–898.

[4] Abdulbasit Darem, Jemal Abawajy, Aaisha Makkar, Asma Alhashmi, and Sultan Alanazi. 2021. Visualization and deep-learning-based malware variant detection using OpCode-level features. *Future Generation Computer Systems* 125 (2021).

[5] Radare2. 2018. Radare2. http://www.radare.org/r/

[6] Jaiteg Singh, Deepak Thakur, Tanya Gera, Babar Shah, Tamer Abuhmed, and Farman Ali. 2021. Classification and Analysis of Android Malware Images Using Feature Fusion Technique. *IEEE Access* 9 (2021), 90102–90117.

[7] Anton Tsitsulin, Davide Mottin, Panagiotis Karras, Alexander Bronstein, and Emmanuel Müller. 2018. Netlsd: hearing the shape of a graph. In *the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining.*