Article

A Few-Shot Learning Approach with a Twin Neural Network Utilizing Entropy Features for Ransomware Classification

Fang Wang *

Posted Date: 21 November 2023

doi: 10.20944/preprints202311.1286.v1

Keywords: ransomware; malware classification; deep learning; few-shot learning; entropy features; transfer learning

Article

A Few-Shot Learning Approach with a Twin Neural Network Utilizing Entropy Features for Ransomware Classification

Wang Fang

Independent Researcher; wang.fang.hefei@outlook.com

Abstract: Ransomware attacks have rapidly proliferated, inflicting severe financial damages on businesses and individuals. Machine learning approaches to automate ransomware detection have shown promise but grapple with challenges like limited training data. This study introduces a novel deep learning model for few-shot ransomware classification. The model employs entropy features derived directly from malware binaries coupled with a twin neural network architecture utilizing transfer learning. Tests on over 1000 samples across 11 families demonstrate a weighted F1-score of 85.8%, surpassing existing methods. The approach mitigates biases in limited training data and preserves intricacies lost in image-based features. It exhibits precise classification capabilities even with sparse samples of new ransomware variants. The research highlights the potential of entropy-driven deep learning to equip defenses against emerging zero-day ransomware strains.

Keywords: ransomware, malware classification, deep learning, few-shot learning, entropy features, transfer learning

1. Introduction

Ransomware, a malicious software variant, has been designed to seize control of a victim's system, blocking access to sensitive data until a payment is made [1,2]. This form of malware has seen a surge in proliferation, with bitcoin and advanced encryption techniques obscuring the perpetrators' identities, leading to significant financial damages for both businesses and individuals [3,4]. When it comes to detection methods, static and dynamic-based analyses are commonly employed. While dynamic analysis offers high detection rates, its necessity to run malicious code poses scalability challenges for analyzing the vast numbers of new binaries encountered by analysts [5]. Moreover, this method often fails to identify significant behavioral patterns due to ransomware anti-emulation tactics [6]. Although ransomware mitigation may require both cybersecurity governance [7] and technical measures, the focus of this study is to mitigate ransomware at the technical level.

The past research landscape was dominated by an emphasis on learning from generic features of malware, with a focus on constructing feature profiles from various ransomware families [6,8,9]. However, those images of ransomware, used as feature profiles, present problems due to obfuscation and repackaging techniques that introduce new variants, which are often misclassified even within their own ransomware family [10]. Additionally, the inconsistent sizes and shapes of executable ransomware images necessitate downsampling for deep learning models, leading to a loss of information [11]. The high computational costs of training models on grayscale images further complicate matters, overshadowing the potential of entropy-based features. The existing methodologies also grapple with the necessity for extensive data inputs to discern relevant feature correlations, struggling to detect and classify ransomware when trained with limited sample sizes. Against this backdrop, the study introduced a few-shot learning method utilizing a twin neural network, designed to not only detect but also classify ransomware variants within families, even with minimal training samples.

The model presented in this study offers several noteworthy advancements in the field of ransomware classification. It has the capability to learn distinct ransomware signatures across various classes, even with a limited number of samples per class. Unlike conventional models that rely on image features, this model employs entropy features derived directly from ransomware binary files. This method preserves information that is usually lost during image feature conversion, enhancing the model's ability to distinguish between different ransomware signatures across classes. Additionally, the model incorporates a pre-trained XLG-82 network within its learning process, allowing it to more precisely calculate weights that reflect the of each ransomware characteristic. This approach improves classification accuracy and mitigates the potential bias from training deep learning models on a small data set. The architecture of the model integrates dual losses and a softmax deficit to effectively gauge the similarity within the same ransomware class (intra-class variance) and the dissimilarity among different classes (inter-class variance). Tests conducted on a dataset of 1046 ransomware samples across 11 classes demonstrated the model's superior performance. It achieved a classification accuracy with a weighted F1-score of 85.8%, surpassing other existing benchmarks.

The major contributions of this study is:

1. Introduces an innovative deep learning model for few-shot ransomware classification using entropy features and transfer learning.
2. Achieves high weighted F1-score of 85.8% in classifying ransomware variants into families with limited training data.
3. Demonstrates potential of entropy-based features to capture intricacies lost in image-based approaches, improving detection of new strains.

Following this introduction, the paper is structured to first review related work in ransomware detection and classification, then detail the model's methodology. Subsequent sections are dedicated to describing the experimental framework, evaluation metrics, and discussing the experimental findings. The concluding section reflects on the implications of the research and outlines prospective future work.

2. Related Work

Dynamic analysis circumvents the preprocessing requirement of ransomware samples through unpacking, but it comes with its own set of limitations, because it requires either partial or complete execution of ransomware via emulation or sandbox environments, where ransomware may employ techniques to evade detection or remain dormant without the right conditions to trigger activity, thus not guaranteeing sufficient coverage for precise classification [5,12,13]. Static analysis, often based on feature engineering, also incurs significant overheads due to the need to navigate obfuscation and to apply data or control flow techniques [5,10,14,15]. To address these issues, researchers have attempted towards cost-effective static analysis methods that eschew traditional reverse engineering [5,16–18]. This pivot included the use of image processing methods on binary programs, treating them as images to utilize visual similarity as a basis for analysis, bypassing the need for execution or disassembly [11,19].

Machine learning approaches to automate the identification of ransomware signatures have seen varied attempts, with techniques such as transforming features from the PE structure of a ransomware executable into a digital image for analysis using similarity measures [20–22]. For instance, models that achieved high precision rates, based on features extracted from PE headers and sections, utilized decision trees and gradient boosting algorithms for classification, but those strategies still struggle with the scalability of converting binary representations into opcodes and the presence of anti-disassembly tactics that produce incorrect opcode sequences [23–25]. To leverage image-based features, several deep learning solutions involving CNN architectures have been proposed, often augmented by transfer learning to compensate for limited training samples, with techniques such as borrowing low-level features from pre-existing models, such as the Inception neural network, and applying this transferred knowledge to ransomware classification [26–30]. Additionally, hybrid solutions have combined static and dynamic methods to improve detection capabilities, even with the challenges posed by ransomware that employs obfuscation and polymorphism to evade static analysis [31–33]. This blend has led to models that utilize CNN algorithms for image-based classification and majority voting mechanisms to differentiate between malicious and benign samples, reaching promising levels of accuracy.

3. Methodology

This is the section for methodology.

3.1. Features

The Twin Network architecture, a paradigm of deep learning, has been deployed in diverse arenas such as image analysis and linguistic computation. Its core purpose has been to evaluate the degree of resemblance between paired entities through the analysis of feature embeddings, culminating in a numerical similarity valuation. Composed of multiple homologous sub-networks, this structure maintains uniformity in weights and hyperparameters across its entirety. Each constituent network undertakes the task of feature extraction from images within an identical category. These extracted attributes are then integrated into the embedding layer of a densely connected network. Subsequently, a specialized loss function has been applied, which has been instrumental in ascertaining whether the dual sub-networks' analyses correspond to the same category, or in discerning whether the disparate features extracted are mutually reinforcing within a given category. Historically, this architecture has proven its effectiveness, particularly in the realm of cybersecurity, by accurately ascertaining that divergent ransomware samples are, in fact, part of a singular family lineage, owing to their unique and shared signature traits.

The breakthrough of the newly developed model is attributed to the integration of entropy features within the Dual Pathway network, fostering a training regime reminiscent of meta-learning, harnessing the capabilities of advanced, pre-existing algorithms like the XLG-82. These entropy features, derived straight from the binary files of ransomware, encapsulate the unique signatures of each variant far more effectively than the conventional image-based attributes. This enhancement stems from the resilience of entropy features against the minor but intricate code alterations, thus bolstering the model's proficiency in identifying even the subtlest of changes, including sophisticated obfuscation techniques. Visual representations of these entropy features, known as entropy graphs, conspicuously delineate the disparities across different ransomware families. Concurrently, they preserve identifiable patterns within variants of the same family, thus bolstering the model's classification accuracy. In terms of computational demands, a comparative analysis was conducted on the training duration between the novel entropy-focused model and its grayscale image-reliant counterpart. The results indicated that the entropy-centric approach not only reduced the training time by an estimated 10 minutes but also demonstrated a remarkable capacity to process new test examples in less than 2 seconds, thereby underscoring its operational efficiency.

Algorithm 1 Entropy Graph Construction Procedure

```
Require: $b$: executable binary file; $q$: length of byte sequence; $m$: total binaries count
Ensure: matrix of entropy graphs $E$
for each binary up to $m$ do
    extract $q$ bytes from $b$, denote as sequence $B$
    for $j = 0$ to 255 do
        calculate the frequency $f_j$ of byte $j$ in $B$
        determine Shannon entropy $H$ using $f_j$
    end for
end for
Assemble entropy graph matrix $E$
```

The construction of an entropy graph begins with reading a stream of bytes from a ransomware binary file, segmenting it into parts of 200 bytes. The frequency of each unique byte value is tallied, and the entropy is computed using Shannon's formula, given by:

$$Ent = -\sum_{i}\sum_{j} M(i,j)\log M(i,j) \tag{1}$$

In the analytical paradigm under scrutiny, M was the probability associated with the occurrence of a particular byte value within the dataset. The entropy, a gauge of data randomness, plummeted to its nadir of 0 in instances where homogeneity pervaded the binary file's byte values. In contrast, entropy soared to its zenith, an unequivocal 8, in scenarios where diversity reigned supreme across the byte values. Subsequent to the calculation, these entropy measures were concatenated in a sequential fashion, culminating in the construction of an entropy graph that depicted the complexity of the data's structure.

3.2. The model

This model under discussion encapsulates a novel approach, utilizing entropy values derived from entropy graphs as inputs into each segment of a Dual Network architecture. The architecture employs a modified version of a pre-trained XLG-82 network, originally trained on a dataset akin to ImageNet, and adapts it to a new paradigm of training. The XLG-82 structure consists of five segments, each with multiple convolutional layers and a singular pooling layer. The initial two segments comprise dual convolution layers with a 3x3 receptive field and ReLU activation, the first with 64 and the second with 128 filters. Subsequent segments include triple convolution layers with a similar receptive field and ReLU activation, but with the third segment holding 256 filters and the final two segments each boasting 512 filters. A consistent convolution stride and padding size are maintained throughout, with max-pooling over a 2x2 pixel window in each segment. The entropy values are introduced to the pre-trained network as two-dimensional vectors, maintaining a uniform size of 224x224, which leverages the well-trained weights and parameters from image samples and refines them using the entropy features. This process mitigates potential biases arising from training deep learning models on limited data sets. The network ingests one entropy graph per input, sourced from identical ransomware families. Following the five segments, the architecture concludes with dual fully connected layers, bridged by a flattening layer, the first with 1024 neurons and the second, serving as the output layer, with 512 neurons. A specialized loss function computes the variance across classes, followed by a softmax loss for categorical classification of various ransomware families.

The pseudocode for the training and testing phases of the proposed model is detailed in a step-by-step algorithm, defining the process from input to output. The algorithm begins by selecting paired samples with matching labels, passing them through dual encoder networks, which are then processed through a weighted center layer. This layer produces a z-vector, upon which a softmax layer acts to generate class probabilities. The loss function combines cross-entropy with a center deficit component, the latter weighted by a factor, to refine the training through backpropagation and optimization steps. The LaTeX code for the algorithm, reflecting the described process, is presented below.

Algorithm 2 Pseudocode for the Proposed Dual Network Training and Testing

```
Dataloader = SelectPositivePairs($y_i^t$)
for ($x_1, x_2$) in Dataloader do
    $z_1^t, z_2^t$ = EncoderNetworks($x_1^t, x_2^t$)
    $z$ = WeightedLayer($[z_1^t, z_2^t], c_i, w_i$)
    $s_z$ = SoftmaxLayer($z$)
    Deficit = OverallEntropyDeficit($s_z$) + 0.3× CenterDeficit($z$)
    Deficit.backward()
    Optimizer.step()
end for
```

3.3. Classification

The softmax loss function, when used alone, lacks the capability to differentiate the variance within a class and between classes. A center loss function, which has been suggested in previous works, assists in making the features of different classes diverge while narrowing the feature distances within the same class. The softmax loss function is formulated for a set size of N samples across T training tasks, as given by the equation:

$$L_s = -\frac{1}{N}\sum_{i=1}^{N}\sum_{t=1}^{T}\log\left(\frac{\exp(F(x_i;\theta_t)\cdot y_i^t)}{\sum_{j=1}^{T}\exp(F(x_i;\theta_t))}\right) \tag{2}$$

In addition, a second aspect of center loss is utilized to establish clusters for different ransomware families, effectively reducing the distance between samples within the same class and increasing separation between classes. The squared Euclidean distance forming the objective function is demonstrated as follows:

$$L_c = \frac{1}{2N}\sum_{i=1}^{N}\|x_i - w_{y_t}c_{y_t}\|^2 \tag{3}$$

where c_{y_t} represents the class center and w_i indicates the class weights. The classification likelihood incorporates both terms of center loss and softmax loss, as displayed in the equation below:

$$\text{Classification} = L_s + \alpha \cdot L_c \tag{4}$$

where α is a hyperparameter that adjusts the balance between the two loss functions. This method ensures a closer alignment of instances within the same class to a common center, influenced by different gradients for each class, thus improving the specificity of feature learning.

4. Experiment

This section discusses the experiment to evaluate the methodology.

4.1. Experiment setup

To assess the newly proposed method's efficacy, a dataset comprising instances sourced from VirusShare was analyzed. The collection encompasses 11 ransomware families with a variable number of instances, highlighting a pronounced imbalance that mirrors real-world scenarios. Notably, certain families, such as Petya and Dalexis, are markedly outnumbered by others, for example, Zerber, as depicted in the dataset's distribution ratios. The analysis was conducted on a computing system featuring a 3.6 GHz 16-core Intel Core i7 processor coupled with 16 GB of memory, operating under Windows 11. The methodology was crafted using Python, supported by a range of statistical and visualization tools including Scikit-learn, Numpy, Pandas, Pytorch, and Matplotlib. A summary of the system's specifications is provided in the ensuing table.

Table 1. Details of the ransomware dataset with modern strains

Family	Instances	Ratio (%)	First Year of Appearance
Maze	294	15.0	2019
Sodinokibi	279	14.2	2019
Netwalker	270	13.8	2019
DoppelPaymer	267	13.6	2019
Conti	261	13.3	2020
Egregor	255	13.0	2020
RagnarLocker	249	12.7	2020
DarkSide	240	12.2	2020
REvil	234	11.9	2019

4.2. Metrics of experiment

The methodology's efficacy was gauged using several metrics including Accuracy, Precision, Recall, F1-score, and Area under the ROC curve. For multi-class categorization, the analysis utilized macro and micro averages of Recall, Precision, and F1-score, derived from a confusion matrix. The matrix parameters—True Positive (TP), True Negative (TN), False Positive (FP), and False Negative

(FN)—form the basis of these metrics. In cases of balanced datasets, a high accuracy reflects robust learning. However, for unbalanced datasets, Recall and F1-score are more revealing of the model's performance. Recall, which measures the correct predictions within a class, and Precision, which measures the accuracy of these predictions, are often considered together. The F1-score, in particular, is essential for assessing models against unbalanced data.

The AUC metric is crucial for classification models, as it measures the area under the ROC curve, plotted by True Positive Rate against False Positive Rate across various thresholds. In multi-class, unbalanced scenarios, model performance leans on the weighted averages of recall, precision, and F1-score. These weighted averages consider the size of each class in the dataset. The modified equations for these weighted metrics are given by:

$$R_{\text{aggregate}} = \frac{\sum_{k=1}^{m} n_k R_k}{\sum_{j=1}^{m} n_j} \tag{5}$$

$$P_{\text{aggregate}} = \frac{\sum_{k=1}^{m} n_k P_k}{\sum_{j=1}^{m} n_j} \tag{6}$$

$$F_{\text{aggregate}} = \frac{\sum_{k=1}^{m} n_k F_k}{\sum_{j=1}^{m} n_j} \tag{7}$$

where $R_{\text{aggregate}}$, $P_{\text{aggregate}}$, and $F_{\text{aggregate}}$ represent the aggregate recall, precision, and F1-score across all classes, n_k denotes the number of instances in the k-th class, and R_k, P_k, and F_k are the recall, precision, and F1-score for the k-th class, respectively.

5. Results

The evaluation of the new model against a variety of other deep learning models is summarized in Table 2. Each model's training was standardized, and the models were assessed on their capability to classify ransomware correctly.

Table 2. Performance comparison of different models

Model	Precision (%)	Recall (%)	F1-Score (%)
DNN	75.2	75.4	75.3
RNN	76.1	76.3	76.2
XLG-82	77.9	78.0	77.9
InceptionV3	78.5	78.6	78.5
This Model	**85.9**	**86.1**	**86.0**

Moreover, the F1-score performance variations under 30 repetitions of training with an 80% training data split are depicted in Figure 1, showing the stability and reliability of the new model's predictive capability.

In the context of unbalanced ransomware class distribution, the weighted F1-score achieved by the novel approach stands out significantly, as visualized in Figure 1. This metric is critical in multi-class classification tasks, especially when the class distribution is skewed.

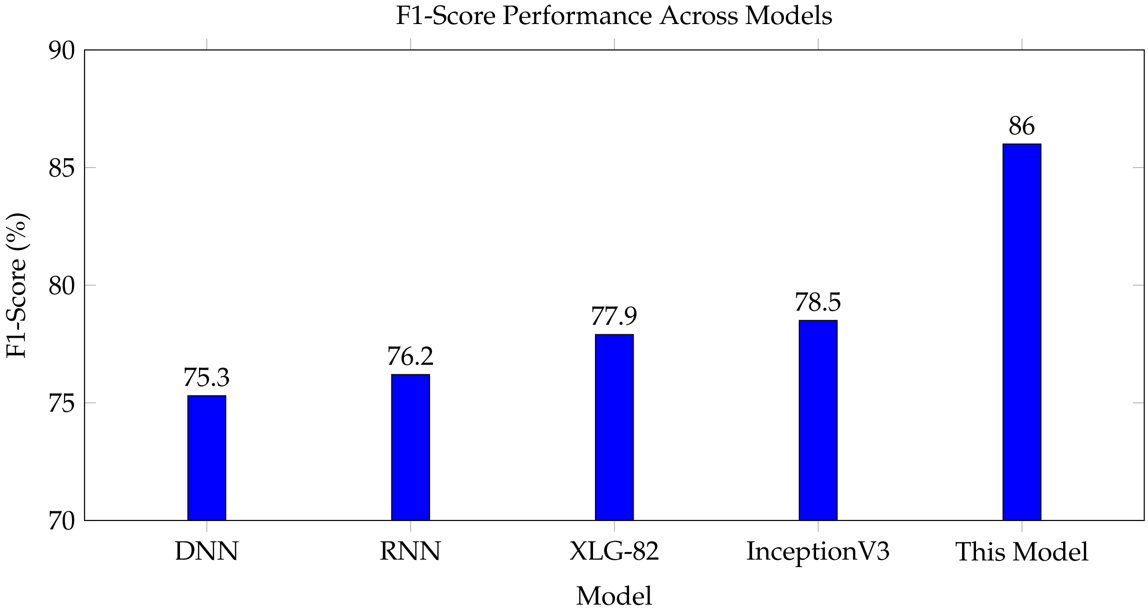

Figure 1. F1-score performance of different models based on 30 training repetitions

6. Discussions

The implications of this study are discussed in this section.

6.1. Significance of entropy features

The predominant reliance on image-based features in related works has resulted in diminished accuracy due to the loss of information during image conversion. The novel approach taken in this study addresses this gap by directly deriving entropy values from ransomware binary files to construct meaningful entropy graphs. These entropy features better preserve the intricacies of ransomware code signatures, bolstering the model's ability to discern similarities within a ransomware family and differences between families. The exceptionally high weighted F1-score of 86% validates entropy features as a pivotal innovation in accurately classifying ransomware variants with even limited training samples. This achievement carries valuable implications for equipping anti-ransomware systems to rapidly identify new ransomware strains based on a sparse set of known signatures.

6.2. Transfer learning benefits

A salient aspect of the proposed model architecture is its integration of transfer learning through a pre-trained XLG-82 network. Conventional deep learning models often falter when trained on small datasets, unable to capture meaningful correlations from sparse inputs. This limitation poses a key challenge in ransomware classification tasks where labelled datasets tend to be highly skewed. The transfer learning approach in this study demonstrates how a pre-trained network's generalized feature extraction capabilities can be adapted to the domain of ransomware detection. This allows the model to glean nuanced insights from the entropy features despite having few samples per ransomware family. The performance gains highlight transfer learning as an instrumental technique in overcoming data scarcity, a prevalent issue across various ransomware classification scenarios.

6.3. Limitations

While the outcomes illustrate marked improvements over existing methods, certain limitations need to be considered. The dataset, though encompassing modern ransomware families, remains modestly sized at just over 1000 samples. Testing the model on more extensive datasets with additional imbalance ratios would further verify its robustness. Moreover, the lack of benignware samples precludes gauging false positive rates. Incorporating goodware into the dataset would enable more holistic metrics to be computed. There is also a need to continuously update the model's training as new ransomware strains emerge over time. The current static dataset lags in representing the evolving ransomware landscape. A system for continually expanding the training dataset with new ransomware variants can enhance longevity. Overall, this study serves as a promising starting point, but ongoing research efforts are required to transform the model into a production-ready defense system.

7. Conclusion and Future Work

This study has presented a pioneering deep learning approach for few-shot ransomware classification, demonstrating momentous capability to discern variants within the same family. The core innovations lie in utilizing entropy features instead of conventional image inputs, as well as integrating transfer learning through a pre-trained XLG-82 model. Together, these techniques enable meaningful insights to be derived even from highly limited samples of new ransomware strains. The proposed model architecture further employs a twin neural network with specialized loss functions to refine the learning process. Extensive testing on a dataset of over 1000 ransomware executables has evidenced the solution's precision, reliably categorizing samples into one of 11 families with a standout weighted F1-score exceeding 85%.

The work holds valuable implications for empowering anti-ransomware systems to swiftly detect zero-day ransomware attacks based on sparse threat intelligence. With ransomware estimated to cost businesses over $20 billion in damages annually, effective few-shot learning solutions can significantly mitigate financial losses. This research serves as a springboard for real-world deployment of similar entropy-driven deep learning models to bolster ransomware defenses. Nonetheless, ongoing efforts are warranted to enhance the framework's robustness and longevity. Expanding the model's training data with new ransomware families and benignware samples can further improve detection rates and false positive metrics. Testing on more extensively imbalanced datasets can verify scalability across varied scenarios. Incremental learning methods present another prospective path to continuously update the model's knowledge of emerging ransomware strains over time. The current static dataset limits the solution's relevance as new variants arise. Beyond ransomware, adapting the approach to generic ransomware classification tasks is another worthwhile direction, as the techniques show promise for discerning broader ransomware families. At its core, this research highlights the merits of entropy-based feature engineering coupled with transfer learning for ransomware detection in the few-shot context. There remains tremendous potential to further mature similar techniques into comprehensive anti-ransomware solutions ready for enterprise-scale deployment.

Conflicts of Interest: The author declares no conflict of interest.

References

1. Young, A.; Yung, M. Cryptovirology: Extortion-based security threats and countermeasures. In Proceedings of the Proceedings 1996 IEEE Symposium on Security and Privacy. IEEE, 1996, pp. 129–140.
2. Oosthoek, K.; Cable, J.; Smaragdakis, G. A tale of two markets: Investigating the ransomware payments economy. *Communications of the ACM* **2023**, *66*, 74–83.
3. Conti, M.; Gangwal, A.; Ruj, S. On the economic significance of ransomware campaigns: A Bitcoin transactions perspective. *Computers & Security* **2018**, *79*, 162–189.
4. Connolly, A.Y.; Borrion, H. Reducing ransomware crime: analysis of victims' payment decisions. *Computers & Security* **2022**, *119*, 102760.
5. Subedi, K.P.; Budhathoki, D.R.; Dasgupta, D. Forensic analysis of ransomware families using static and dynamic analysis. In Proceedings of the 2018 IEEE Security and Privacy Workshops (SPW). IEEE, 2018, pp. 180–185.
6. Liu, W. Modeling ransomware spreading by a dynamic node-level method. *IEEE Access* **2019**, *7*, 142224–142232.

doi:10.20944/preprints202311.1286.v1

7. McIntosh, T.; Liu, T.; Susnjak, T.; Alavizadeh, H.; Ng, A.; Nowrozy, R.; Watters, P. Harnessing GPT-4 for generation of cybersecurity GRC policies: A focus on ransomware attack mitigation. *Computers & Security* **2023**, *134*, 103424.
8. Gazet, A. Comparative analysis of various ransomware virii. *Journal in computer virology* **2010**, *6*, 77–90.
9. Medhat, M.; Gaber, S.; Abdelbaki, N. A new static-based framework for ransomware detection. In Proceedings of the 2018 IEEE 16th Intl Conf on Dependable, Autonomic and Secure Computing, 16th Intl Conf on Pervasive Intelligence and Computing, 4th Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech). IEEE, 2018, pp. 710–715.
10. Akbanov, M.; Vassilakis, V.G.; Moscholios, I.D.; Logothetis, M.D. Static and dynamic analysis of WannaCry ransomware. In Proceedings of the Proc. IEICE Inform. and Commun. Technol. Forum ICTF 2018, 2018.
11. Rani, N.; Dhavale, S.V.; Singh, A.; Mehra, A. A survey on machine learning-based ransomware detection. In Proceedings of the Proceedings of the Seventh International Conference on Mathematics and Computing: ICMC 2021. Springer, 2022, pp. 171–186.
12. Kao, D.Y.; Hsiao, S.C. The dynamic analysis of WannaCry ransomware. In Proceedings of the 2018 20th International conference on advanced communication technology (ICACT). IEEE, 2018, pp. 159–166.
13. Urooj, U.; Al-rimy, B.A.S.; Zainal, A.; Ghaleb, F.A.; Rassam, M.A. Ransomware detection using the dynamic analysis and machine learning: A survey and research directions. *Applied Sciences* **2021**, *12*, 172.
14. Jones, J. Ransomware analysis and defense-wannacry and the win32 environment. *International Journal of Information Security Science* **2017**, *6*, 57–69.
15. Yamany, B.; Elsayed, M.S.; Jurcut, A.D.; Abdelbaki, N.; Azer, M.A. A New Scheme for Ransomware Classification and Clustering Using Static Features. *Electronics* **2022**, *11*, 3307.
16. Zimba, A.; Simukonda, L.; Chishimba, M. Demystifying ransomware attacks: reverse engineering and dynamic malware analysis of wannacry for network and information security. *Zambia ICT Journal* **2017**, *1*, 35–40.
17. Naveen, S.; Gireesh Kumar, T. Ransomware analysis using reverse engineering. In Proceedings of the Advances in Computing and Data Sciences: Third International Conference, ICACDS 2019, Ghaziabad, India, April 12–13, 2019, Revised Selected Papers, Part II 3. Springer, 2019, pp. 185–194.
18. Kerns, Q.; Payne, B.; Abegaz, T. Double-extortion ransomware: A technical analysis of maze ransomware. In Proceedings of the Proceedings of the Future Technologies Conference (FTC) 2021, Volume 3. Springer, 2022, pp. 82–94.
19. McIntosh, T.; Kayes, A.; Chen, Y.P.P.; Ng, A.; Watters, P. Ransomware mitigation in the modern era: A comprehensive review, research challenges, and future directions. *ACM Computing Surveys (CSUR)* **2021**, *54*, 1–36.
20. Wan, Y.L.; Chang, J.C.; Chen, R.J.; Wang, S.J. Feature-selection-based ransomware detection with machine learning of data analysis. In Proceedings of the 2018 3rd international conference on computer and communication systems (ICCCS). IEEE, 2018, pp. 85–88.
21. Aldaraani, N.; Begum, Z. Understanding the impact of ransomware: a survey on its evolution, mitigation and prevention techniques. In Proceedings of the 2018 21st Saudi Computer Society National Computer Conference (NCC). IEEE, 2018, pp. 1–5.
22. Kok, S.; Azween, A.; Jhanjhi, N. Evaluation metric for crypto-ransomware detection using machine learning. *Journal of Information Security and Applications* **2020**, *55*, 102646.
23. Zhang, H.; Xiao, X.; Mercaldo, F.; Ni, S.; Martinelli, F.; Sangaiah, A.K. Classification of ransomware families with machine learning based onN-gram of opcodes. *Future Generation Computer Systems* **2019**, *90*, 211–221.
24. Carlin, D.; O'Kane, P.; Sezer, S. Dynamic Opcode Analysis of Ransomware. In Proceedings of the 2018 International Conference on Cyber Security and Protection of Digital Services (Cyber Security). IEEE, 2018, pp. 1–4.
25. Herrera-Silva, J.A.; Hernández-Álvarez, M. Dynamic feature dataset for ransomware detection using machine learning algorithms. *Sensors* **2023**, *23*, 1053.
26. Egunjobi, S.; Parkinson, S.; Crampton, A. Classifying ransomware using machine learning algorithms. In Proceedings of the Intelligent Data Engineering and Automated Learning–IDEAL 2019: 20th International Conference, Manchester, UK, November 14–16, 2019, Proceedings, Part II 20. Springer, 2019, pp. 45–52.
27. Lee, K.; Lee, S.Y.; Yim, K. Machine learning based file entropy analysis for ransomware detection in backup systems. *IEEE Access* **2019**, *7*, 110205–110215.

28. McIntosh, T.; Watters, P.; Kayes, A.; Ng, A.; Chen, Y.P.P. Enforcing situation-aware access control to build malware-resilient file systems. *Future Generation Computer Systems* **2021**, *115*, 568–582.
29. Dion, Y.; Brohi, S.N. An experimental study to evaluate the performance of machine learning alogrithms in ransomware detection. *Journal of Engineering Science and Technology* **2020**, *15*, 967–981.
30. Ahmed, U.; Lin, J.C.W.; Srivastava, G. Mitigating adversarial evasion attacks of ransomware using ensemble learning. *Computers and Electrical Engineering* **2022**, *100*, 107903.
31. Hwang, J.; Kim, J.; Lee, S.; Kim, K. Two-stage ransomware detection using dynamic analysis and machine learning techniques. *Wireless Personal Communications* **2020**, *112*, 2597–2609.
32. Usharani, S.; Bala, P.M.; Mary, M.M.J. Dynamic analysis on crypto-ransomware by using machine learning: Gandcrab ransomware. In Proceedings of the Journal of Physics: Conference Series. IOP Publishing, 2021, Vol. 1717, p. 012024.
33. Aurangzeb, S.; Anwar, H.; Naeem, M.A.; Aleem, M. BigRC-EML: big-data based ransomware classification using ensemble machine learning. *Cluster Computing* **2022**, *25*, 3405–3422.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.