Computer Modeling in
Engineering & Sciences

Tech Science Press

Doi:10.32604/cmes.2025.058352

ARTICLE

MAD-ANET: Malware Detection Using Attention-Based Deep Neural Networks

Waleed Khalid Al-Ghanem[1], **Emad Ul Haq Qazi**[2,*], **Tanveer Zia**[2,3], **Muhammad Hamza Faheem**[2], **Muhammad Imran**[4] **and Iftikhar Ahmad**[5]

[1]Department of Computer Engineering, College of Computer and Information Sciences, King Saud University, Riyadh, 11362, Saudi Arabia

[2]Centre of Artificial Intelligence, Naif Arab University for Security Sciences, Riyadh, 14812, Saudi Arabia

[3]School of Arts and Sciences, The University of Notre Dame, Sydney, NSW 2007, Australia

[4]Center for Smart Analytics, Institute of Innovation, Science and Sustainability, Federation University Australia, Berwick, VIC 3806, Australia

[5]Department of Information Technology, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, 21589, Saudi Arabia

*Corresponding Author: Emad Ul Haq Qazi. Email: qabdulrab@nauss.edu.sa

Received: 10 September 2024; Accepted: 06 March 2025; Published: 11 April 2025

ABSTRACT: In the current digital era, new technologies are becoming an essential part of our lives. Consequently, the number of malicious software or malware attacks is rapidly growing. There is no doubt, the majority of malware attacks can be detected by most antivirus programs. However, such types of antivirus programs are one step behind malicious software. Due to these dilemmas, deep learning become popular in the detection and classification of malicious data. Therefore, researchers have significantly focused on finding solutions for malware attacks by analyzing malicious samples with the help of different techniques and models. In this research, we presented a lightweight attention-based novel deep Convolutional Neural Network (DNN-CNN) model for binary and multi-class malware classification, including benign, trojan horse, ransomware, and spyware. We applied the Principal Component Analysis (PCA) technique for feature extraction for binary classification. We used the Synthetic Minority Oversampling Technique (SMOTE) to handle the imbalanced data during multi-class classification. Our proposed attention-based malware detection model is trained on the benchmark malware memory dataset named CIC-MalMem-2022. The results indicate that our model obtained high accuracy for binary and multi-class classification, 99.5% and 97.9%, respectively.

KEYWORDS: Attention-based CNN; malware detection; machine learning; deep learning; classification

1 Introduction

Nowadays, the quick advancements in technology have impacted both personal daily life practices and business operations. Additionally, in the cyber world, privacy is an essential right of users that must be secure via technological efforts. Cyberattacks have become a common threat that can damage the data or sensitive information of individual user enterprises or even governments. These actions can be performed by attackers who utilize malware or harmful software to get unauthorized access to the user's system, steal confidential information of users for financial gains, or damage their system and make them vulnerable. Malicious software (malware) is a computer program that gets different names, such as ransomware, worms, spyware, and trojans, that are designed to use or harm the operating system (OS), networks, or devices [1,2].

Moreover, there are multiple common cyberattacks, such as web application attacks to get unauthorized access to the user's private data, DoS attacks to disable user's systems, services, or networks and make them unavailable, and many more. Each category of malicious software with specialized families has its own architecture and working mechanisms. Currently, the primary malware categories including Spyware, Trojan horses, and Ransomware retain one shared capability which allows them to operate undetected by security protocols until they achieve complete or partial completion of their attack objective [3].

In [4], the authors proposed an advanced deep learning system to detect malware in healthcare IoT devices and smartphones. This approach utilizes neural networks to analyze and categorize malware threats effectively. The system achieved impressive results on benchmark datasets, showcasing its potential to enhance the security of IoT and smartphone environments. Furthermore, each malware sub-category has multiple functionalities, for instance, gaining access to sensitive information or carrying out other cybercrimes within each category, the efficient solution for its detection should focus on each category along with its families to prevent and stop them in the future. Multiple learning systems, such as deep learning and machine learning, are utilized instead of manual detection methods due to their high complexity and time consumption. These systems may automatically extract intelligent insights from the data. These learning systems are mostly useful for finding appropriate training data to feed the system and make a more accurate evaluation. Machine learning systems get a set of attributes that can be examined in detail to compare and contrast differences [5].

Another study [6] designed a spatial attention and convolutional neural network (SACNN) framework that applied deep learning techniques to classify 25 malware families using images with or without balanced class distributions. They performed experiments on the Malimg benchmark dataset and achieved 97.42% precision, 97.95% recall, 97.11% precision, and 97.32% F1-score, respectively. Additionally, there are several formats in which features can be inputs, which is a factor in finding the machine learning system that should be utilized. Whereas certain ML algorithms emphasize speed, others focus on precision and accuracy. Consequently, choosing the different algorithms that match the objective and type of input has a significant impact on the outcome of the system [7,8].

In machine learning, the ensemble model is well correlated with these detection and characterization objectives. Multiple existing malware detection techniques often struggle with hidden or new types of malware, as well as with imbalanced data. These approaches mainly focused on identifying non-obfuscated privacy malware that makes them hard to use in real time. However, existing programming mechanisms cannot infer a general rule from the patterns and behaviors of various families and categories of obfuscated privacy malware, which shows the need for efficient, fast, more adaptable solutions. Therefore, machine learning algorithms, specifically deep neural networks, are frequently utilized for analyzing privacy-related malware [9].

Malware that hides its features, functions, and activities as its central technique of eluding security measures during runtime is referred to as obfuscated malware. It is difficult to address malware that has been obfuscated because it resists techniques based on signature analysis used by security controls, including intrusion detection systems, antivirus, and intrusion protection system (IPS) engines [10]. Detection metrics against non-obfuscated malware are higher. However, this resistance leads to less effective ones. Furthermore, it is quite difficult to handle multiclass classification when dealing with obfuscated malware since it prevents the establishment of features that allow a distinct division between various malware families and categories [11]. It is essential to identify the type of privacy malware being attacked to put appropriate security controls and countermeasures in place and comprehend the nature of attacks [12]. Multiple strategies have been proposed to collect data that enable the detection and categorization of obfuscated privacy malware. Such as the investigation of Domain Generation Algorithms (DGA) [13], the identification of patterns in command-and-control sequences, and the examination of DNS patterns to identify malware hosting patterns [14]. With the use of 6G and other emerging technologies, cybersecurity faces new complexities. These systems introduce huge data volumes with very low latency and a maximum number of connected devices, all of which expand the attack surface and produce new vulnerabilities. Malware detection models should be able to handle these challenges by becoming faster and more capable of using high-speed networks.

Unfortunately, the majority of the work has significant levels of complexity and time consumption, which makes them unsuitable for real-world applications. This is the motivation to present a lightweight, efficient, and fast model using an attention-based deep neural network architecture for obfuscated malware that is optimized with principal component analysis (PCA) for binary classification and SMOTE for multiclass. The major contributions of this research include:

- We propose a lightweight attention-based novel Convolutional Neural Network (CNN) model for binary classification of obfuscated malware, the benign and malicious respectively that uses an attention layer, flattered layer, dense layer, and multiclass obfuscated malware classification including benign, trojan horse, ransomware, and spyware to gain high metrics and computational efficiency.
- We evaluate our proposed approach using different performance metrics to ensure the reliability of the proposed attention-based CNN model with the existing machine-learning techniques.
- A detailed comparison between the proposed approach and traditionally used approaches is also presented to depict the robustness of the binary and multiclass malware detection architecture.
- A detailed analysis of the CIC-MalMem-2022 [15] dataset is also presented in this research.

The subsequent sections follow this structure. Section 2 represents the multiple existing studies that were conducted by various researchers on obfuscated malware detection. Section 3 presents the proposed technique and describes the dataset named CIC-MalMem-2022, as well as the experimental setup in detail. Section 4 describes the results and analysis of the research, and evaluation and comparison with existing approaches are carried out in Section 5. Finally, Sections 6 and 7 contain research limitations and future work.

2 Literature Review

Since the establishment of malware, it has obtained a lot of attention in the cyber-security field because of its multiple delivery methods and categories. While there are multiple detecting techniques, each approach has its own set of difficulties. In existing studies, mostly utilized different deep learning techniques, including convolutional networks and recurrent, while some of them used machine learning techniques. This section explains the related works on malware detection and addresses the limitations and challenges of these techniques.

In [16], the authors performed binary classification along with the big data technique for malware detection by using the CIC-MalMem-2022 dataset that contains data of memory analysis. They used Apache Spark big data forum to introduce a new direction for malware detection based on memory analysis and established a classification model using traditional nine deep learning and machine learning algorithms. With an accuracy rate of 99.97%, the Logistic Regression (RL) proved to be the most efficient algorithm among them. However, they used the complex deep neural network architecture to tackle the binary classification problem. In another study [14], the authors proposed a structural variations approach by using the Bidirectional Long Short-Term Memory (BiLSTM) method to enhance the time performance of classifiers. The main focus of this study was on analyzing behaviors of dynamic requests for detecting and classifying polymorphic malware by using the Windows API Calls Sequences dataset.

An attention-based multi-dimensional deep learning (DL) approach has been proposed in [17] for a cross-architecture IoMT malware detection and classification system. The presented approach was based on byte sequences extracted from Executable and Linkable Format files that automate the feature extraction process from unstructured byte sequences. They used the IoMT cross-architecture benchmark dataset for IoMT malware detection and gained 94% accuracy.

To identify the malicious DNS found in volatile memory that leads to malware, the authors proposed a Heterogeneous Deep Neural Network (HDNN) [13] that is capable of executing trojan horses, and botnets carried out DDoS attacks and stealthy exploits, respectively. The 360netlab DGA dataset was used in this research and gained a high accuracy rate for trojan horses and botnet threats. However, this research only focused on DNS name recognition produced by malware. Similar research was conducted in [18–20], but their method was based on traditional ML algorithms used in the cloud-based services environment. The created system can detect and classify stealthy malware attacks by utilizing a dataset created through memory introspection techniques.

In other studies [21,22], the authors used the advantage of NLP (Natural Language Processing) methods as a baseline to get the local spatial correlations and presented a CNN-LSTM model by combining the neurons of both CNNs and LSTM models that aim to detect the malware in real-time. Moreover, the author split the dataset into two sets, one set for training and the second set for testing the proposed trained model. As a result, the presented model gained the highest accuracy, at 99% among the decision tree (DT) and support vector machine (SVM) algorithms respectively.

In [23], the authors used deep learning and machine learning to enhance the process of malware detection to identify malware accurately without any manual interference, which often leads to overlooking obfuscated malware. They train and validate multiple deep-learning neural networks, including convolutional and recurrent neural networks while visualizing the malware for better analysis simultaneously. They achieved over 98% validation accuracy using CNN. However, the drawback of this research is that it only focused on visualizing malware techniques. The study [24] performed the binary classification by using the memory data in trojan horse detection. The authors used the decision tree classifier with memory data and experiments were performed on the CIC-MalMem-2022 dataset and as a result, gained 98.41% accuracy.

The dilated convolutional neural network proposed in [25,26] classified the memory dump data with obfuscated malware. The presented algorithm deals with the obfuscated malware detection problem in memory. They used the latest CIC-MalMem-2022 dataset which considers the current era of technologies. Additionally, this research provides different techniques for malware detection and classification from the existing memory information as well as tested traditional methods of machine learning optimization approaches along with hyper-parameters. The proposed neural network architecture achieved 83% accuracy while classifying the malware family.

The authors presented a new model [27] for APT attack detection built on the Adversarial Tactics Techniques and Common Knowledge (ATT&CK) matrix named Strange Behavior Inspection (SBI). The aim of introducing this model is to detect the APT on the first potential victim when the attackers utilize the credential dumping technique. This study presented results level by level, such as first is random access memory, second is CPU, third is Windows registry, and last is file.

In another research [28], the Running Window Entropy (RWE) based malware classifier named Malgazer was designed and developed to achieve the best results. In this research, 60,000 malware samples including trojan horses, PUA, backdoors, ransomware, worms, and viruses are used. They evaluated eight different machine learning algorithms using the RWE and the GIST features in this study. The proposed classifier gains approximately 76% accurate results as compared to the other leading classifiers. In [29], the authors proposed an end-to-end AEFETA framework that is based on the self-DNN to handle the encrypted network traffic classification task. The proposed framework extracted features and patterns from traffic seen through PCAP (raw packet capture) files. In another study [30], the authors used the Windows Performance Counters dataset to detect and classify the crypto-mining malware that allows it to take sensitive data and particular types of ransomware. This research gathers data through PowerShell analysis. The operating Context Profiling system proposed in [31] is based on deep neural networks LSTM networks. The Windows Performance Counters data is used for detecting crypto-mining applications. In [32], the authors proposed a deep learning-based efficient malware detection model. The presented model used the reweighted class balanced loss function while handling the imbalanced data to obtain crucial performance improvements in malware classification. In another study [33], a new hybrid CNN model has been proposed for malware classification. This research developed a B2IMG approach to transform the byte files to RGB and gray image formats. The authors presented a new CycleGAN-based data augmentation system to tackle the inconsistent data size problem between malware families.

In [34], the authors presented a malware detection and classification approach that used memory forensic techniques to extract memory-based features from memory images. The extracted features can show the behavior of malware [35], for instance, linking with the operating system, interacting with commands and control site, DLL, and process injection. Before training and testing the proposed classifier, the feature engineering technique was used for feature extraction and conversion of these features to binary vectors. In another study [36], the authors proposed the MaIFCS malware classification framework to visualize the malware binaries on structural entropy as entropy graphs. They used the deep CNN model to extract features and SVM to classify the malware patterns. Stiawan et al. [37] use opcode behavior analysis with the k-Nearest Neighbors (k-NN) algorithm to detect ransomware. It achieves high detection accuracy by distinguishing between malicious and legitimate software through opcode sequence patterns. The approach is lightweight and effective for malware classification. In [38], the authors propose a stacked ensemble model combining multiple machine-learning techniques to improve intrusion detection in cloud systems. It identifies attacks like DDoS and unauthorized access, offering a scalable and real-time solution for cloud security. The study is tested on real-world datasets. Cevallos-Salas et al. [39] focus on identifying obfuscated malware using memory dumps to extract key features. The method outperforms traditional signature-based detection and highlights the importance of memory-based analysis in modern malware detection.

Detecting and classifying the multiclass problem among the malware categories is not performed in [7] nor [16] as shown in Table 1. However, the authors highlighted that the CIC-MalMem-2022 dataset's observations contain an analysis of in-memory obfuscated malware that is difficult to examine and classify into different categories with standalone classifiers. In other studies [12], and [22], respectively, the multiclass classification problem was analyzed by utilizing the CIC-MalMem-2022, but the proposed solutions are only based on complex CNN architectures. Deep learning models have attention mechanisms that help us emphasize the most significant features of the data, which improves our ability to identify small indications of malicious behavior. So, our study focused on the detection and classification of binary as well as multiclass obfuscated malware detection and enabled it to differentiate whether a process gives general patterns and behaviors of a malicious or benign infection as well as This will help to contribute to more secure computing environments by enhancing the ability to respond against malware threats.

Table 1: Literature review comparison

Reference	Dataset	Problem	Methodology	Data gathering	Limitation
[7]	CIC-MalMem-2022	Binary classification	Used traditional ML Algorithms	Memory dumping	Mainly focus on binary classification problems by using the ML models
[12]	CIC-MalMem-2022	Multiclass classification	Convolutional neural networks	Memory dumping	Use of complex CNNs for multiclass classification problem
[14]	Windows API Calls Sequences	Binary classification	BiLSTM	For binary classification. limited to dynamic requests behavior analysis using API patterns	For binary classification, limited to dynamic requests behavior analysis
[16]	CIC-MalMem-2022	Binary classification	Used traditional ML algorithms and DNNs	Memory dumping	Focus on binary classification problems using the Complex DNN architecture
[13]	360netlab DGA	Multiclass classification	HDNN	Used DNS patterns	Focus on only malicious DNS identification
[18]	University of Mexico (UNM) and BareCloud datasets	Multiclass classification	Random forest	Used hardware patterns	Limited to obfuscated malware families specialized in attacking virtual Domains
[22]	CIC-MalMem-2022	Multiclass classification	Convolutional neural networks	Memory dumping	Use of complex CNNs for multiclass classification Problem

(Continued)

Table 1 (continued)

Reference	Dataset	Problem	Methodology	Data gathering	Limitation
[23]	SBI dataset	Multiclass classification	Introduced Strange Behavior Inspection (SBI)	Used APT patterns	Focus on only APT attack classification
[28]	RWE	Multiclass classification	Used traditional ML algorithms	Performed entropy analysis	The proposed classifier does not comprise to Spyware
[29]	PCAP dataset	Multiclass classification	Self-DNN based AEFETA framework	Performed PCAP analysis	Focused on ransomware cyphering malware against privacy
[30]	Windows performance counters dataset	Multiclass classification	DNN-LSTM	Performed powershell analysis	Limited to crypto mining malware and able to get sensitive data and detect only ransomware types
[31]	Spamdataset2	Multiclass classification	CNN-LSTM	Used phishing patterns	Limited to particular types of obfuscated privacy malware roll out via emails, URLs, and spamming
Proposed work	CIC-MalMem-2022	Binary and Multiclass classification	Attention-based DNN-CNN	Memory dumping	Propose a novel model to tackle the previously mentioned limitations in the literature.

3 Methodology

This section contains a detailed explanation of the used dataset named CIC-MalMem-2022 (Section 3.1), a preprocessing strategy that was applied to the dataset (Section 3.2), data splitting for training and testing (Section 3.3), proposed methodology with detailed description (Section 3.4), utilized performance metrics for experiments (Section 3.5) and experimental setup (Section 3.6). Fig. 1 demonstrates the methodology overview for our research presentation consisting of three distinct phases. The first phase is dataset preparation by data cleaning and preprocessing to set the foundation for the obfuscated malware detection model in the next phase. This phase comprised the implementation of the Attention-based DNN-CNN model for binary and multiclass classification. The third phase obtained the results, comparison, and conclusion.

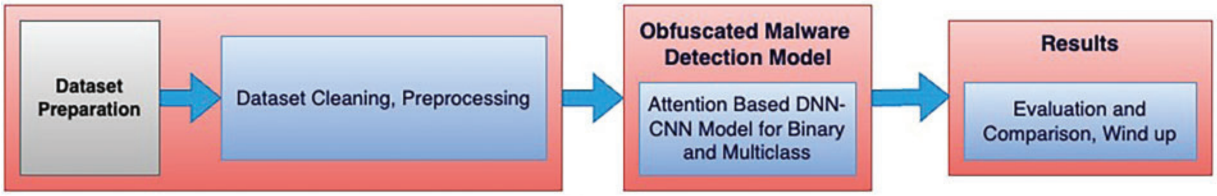
Dataset Preparation
Dataset Cleaning, Preprocessing
Obfuscated Malware Detection Model
Attention Based DNN-CNN Model for Binary and Multiclass
Results
Evaluation and Comparison, Wind up

Figure 1: Overview of proposed methodology

3.1 Dataset

This study utilized the Malware Memory Analysis CIC-MalMem-2022 dataset [15] which was originated by the Canadian Institute for Cybersecurity of the University of New Brunswick (CIC-UNB) in 2022. The CIC-MalMem-2022 is a modern dataset-based memory dumping analysis and contains information on whether it is malware or not. It has a total of 58,596 records that contain 50% benign memory dumps and 50% malicious memory dumps as shown in Table 2.

Table 2: CIC-MalMem-2022 dataset distribution

Dataset distribution	No. of instances
Malware	29,298
Benign	29,298
Total	**58,596**

This dataset not only contains benign and malware classes but is also made up of different categories of malware that are shown in Fig. 2, such as Trojan Horse, Ransomware, and Spyware malware as shown in Table 3.

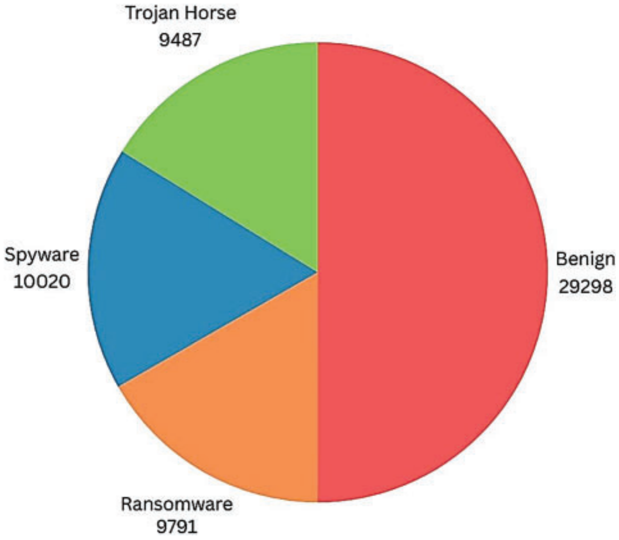

Figure 2: Overview of CIC-MalMem-2022 dataset distribution

Table 3: CIC-MalMem-2022 dataset distribution

Dataset distribution	Malware families	No. of instance
Couti		1988
Ransomware	Maze	1958
Pysa		1717
Ako		2000
Shade		2128
SpywareCoolwebsearch	180Solutions	2000
Gator		2000
Transponder		2200
TIBS		2410
Trojan Horse	Zeus	1950
Emotet		1967
Refroso		2000
Scar		2000
Reconyc		1570
Total		**29,298**

3.2 *Preprocessing*

We performed transformations on the data to prepare it for classification-based analysis. These steps are important to enhance the classification model performance. Additionally, it shows data in an appropriate format for the use of deep learning and machine learning algorithms. Moreover, the used CIC-MalMem-2022 dataset is balanced and contains two classes, malware and benign respectively. Therefore, no data imbalance was performed for the overfitting problem while binary classification used the PCA technique to get the main features in a dataset. However, for multiclass detection and classification, the SMOTE data balancing technique is carried out to overcome the overfitting issues posed by random oversampling, mainly in the deep learning approach. This process also prevents the unessential consumption of resources.

3.3 *Splitting*

In this research, we split the dataset in a stratified way between the types for training and testing for the proposed model. We used 80% of the total data for training and 20% for testing.

3.4 *Proposed Methodology*

This study organized its methodology into distinct phases as presented in Fig. 3. The information collected at each sub-stage of the process combined to establish the study's final research conclusion. The first phase involved preparing the dataset through the acquisition of patterns and behaviors from the three types of obfuscated malware. Then preprocessing was performed on the input CIC-MalMem-2022 dataset. In this phase, we performed feature selection and data normalization on the acquired dataset that enabled recognition and differentiating among the used dataset observations corresponding to the malicious and benign memory dumps (for binary classification).

Additionally, the three sub-categories of obfuscated privacy are malicious (for multiclass classification). We also applied the PCA technique to standardize and dimensionally minimize training data to two axes for binary classification and used SMOTH to tackle the imbalanced data in multiclass obfuscated malware. In the second phase, we applied a convolutional block that contains filters that are used for detecting the specific features or patterns in the input dataset.

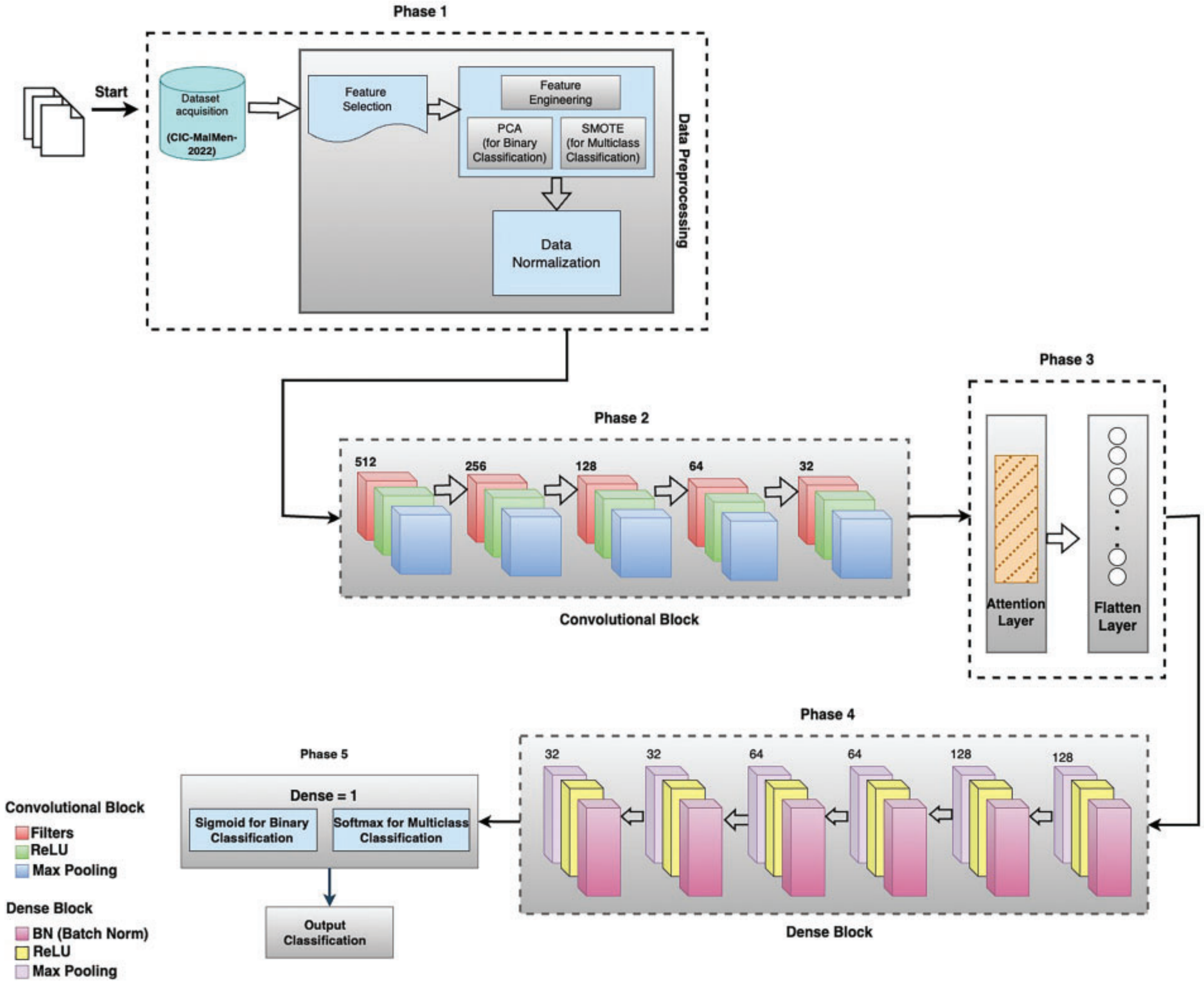

Figure 3: Proposed attention-based DNN-CNN model for binary and multiclass malware detection

Then, the rectified linear unit (ReLU) function replaces all the negative pixel values in the feature map with zero to show non-linearity. Furthermore, applied max pooling to extract the maximum values from the feature map stated by the filter size and strides. In phase three, we used the attention layer to improve the performance of the model to capture long-range dependencies and handle complex tasks. It also allows us to selectively focus on particular parts of the input sequence while predicting and generating output. The attention layer is used to calculate the attention weights, and these weights are utilized to weigh the contributions of multiple elements when generating the model's output. Along with the 1 attention layer, we used the 1 flatten layer that feeds from the output of the convolutional block. Next, in phase four, we used a dense block that consists of three layers 32, 64, and 128 respectively, and contains 3 batch normalization layers (1 per dense layer), ReLU activation function, and max pooling layers (1 per dense layer) respectively. The dense block connects all layers directly with each other and every layer obtains additional inputs from all previous layers. At the last phase, we used the "Sigmoid" activation function which is an ideal logistic function for binary classification and the "SoftMax" activation function for multiclass classification which enabled us to achieve a vector of probabilities for each class analyzed. We used these activation functions to increase the performance of our proposed model while training. We performed 50 epochs to train our presented model using an initial learning rate of 0.001, which was reduced by a factor of 0.1 if validation performance plateaued. For overfitting prevention, we implemented the early stopping with a patience of 5 epochs. Finally, our proposed attention-based DNN-CNN model achieved the highest classification accuracy rate of 99.5% for binary classification and 97.9% for multiclass classification.

3.5 Performance Metrics

This research optimized the model results to evaluate the performance of the Attention-based DNN-CNN model. For that purpose, we used different evaluation metrics. The main metric of interest was accuracy, which computed the proportion of the correctly classified number of samples from the CIC-MalMen-2022 dataset. Meanwhile, the comprehensive evaluation included calculating precision, recall, and F1-score along with other performance metrics. Mathematically, we can describe these metrics as:

(a) Accuracy

The number of correct true positive predictions represents accuracy, measured as a percentage of total predicted outcomes:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

(b) Precision

Precision describes how many correct predictions of positive cases exist among all positive predictions:

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

(c) Recall

The proportion of correct positive predictions among all operational outcomes indicates recall value:

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

(d) F1-score

F1-score is the harmonic mean of recall and precision, representing a balanced measure of the model performance:

$$F1 - score = 2\times = \frac{precision.recall}{precision + recall} = \frac{TP}{TP + \frac{1}{2}(Fp + FN)} \tag{4}$$

where,

True Positive (TP) = Number of true-positive rates that have been correctly identified

True Negative (TN) = Number of true-negative rates that have been incorrectly identified

False Positive (FP) = Number of false-positive rates that have been incorrectly identified

False Negative (FN) = Number of false-negative rates that have been correctly identified

3.6 Experimental Setup

As we described earlier, we divided our dataset for training and testing with the “train_test_split” set into an 80:20 ratio. We carried out experiments on the Google Collaboratory environment as we described in Table 4.

Table 4: Overview of experimental setup

Environment	Resources
Environment	Google colaboratory
Disk capacity	78.2 GB
Programming language	Python 3.7
RAM	12.7 GB
GPU	T4 GPU

4 Result and Analysis

In this section, we have provided the results and a complete analysis of the results achieved with the presented methodology. We trained our Attention-based DNN-CNN proposed model using our trained data. We have performed two experiments on the same dataset, and the first experiment was carried out to accurately perform the binary classification. For that purpose, we used the PCA technique to minimize the training data to two axes and the Sigmoid activation function to map the input values between 0 and 1.

The second experiment was performed for multiclass classification. That includes benign and malware families, such as Spyware gathering data related to the user’s browsing activities and conveying it to third parties. Ransomware focuses on getting funds directly from the user and restricts their access by encrypting files, drives, or other data on the system. The Trojan horse carries out malicious software activities in the background, but it is harmless to users. We addressed the imbalance data through the SMOTH technique while performing multiclass classification and applied the SoftMax activation function to achieve a vector of probabilities for each class analyzed.

4.1 Binary Malware Classification Using Attention-Based DNN-CNN Model

Fig. 4 indicates the resulting confusion matrix to evaluate the proposed model for binary classification on the CIC-MalMem-2022 dataset. According to this figure, 26 observations were classified as FP cases that influence accuracy and precision, while 22 were classified as FN cases that influence recall and, consequently, the F1-score. These observations indicate that our model can minimize the FN rate to obtain the high recall matrix. As we previously described, we used the PCA technique for binary classification, which helped achieve the high matrix and maintained the balance in the input dataset among the malicious and benign observations which made the classification problem easy considering both classes equally. The results indicate that our model achieved the highest accuracy 99.5% for binary classification, as shown in Table 5.

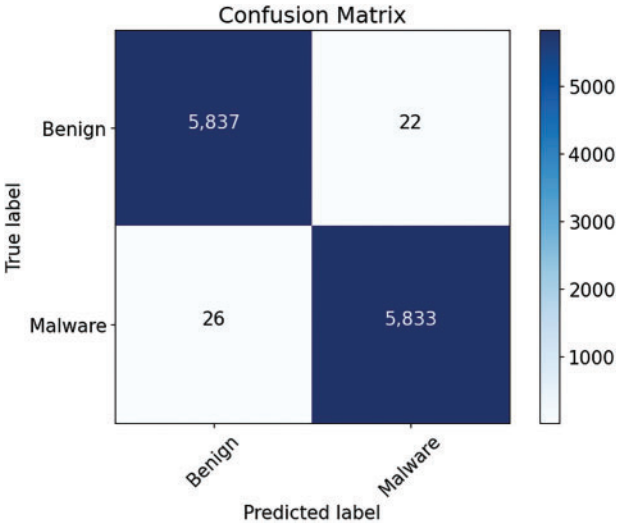

Figure 4: Binary classification confusion matrix

Table 5: Evaluation metrics of binary and multiclass classification

Metrics	Accuracy	Precision	Recall	F1-score
Binary classification	99.5%	99.6%	99.5%	99.5%
Multiclass classification	97.9%	97.9%	98.0%	97.9%

4.2 Multiclass Malware Classification Using Attention-Based DNN-CNN Model

The obtained results indicate that our Attention-based DNN-CNN model leverages SMOTE techniques and SoftMax activation function, strategically comprising deep learning concepts. According to the provided Fig. 5. of the confusion matrix, we achieved 97.9% accuracy on multiclass classification, and other obtained metrics are listed in Table 5.

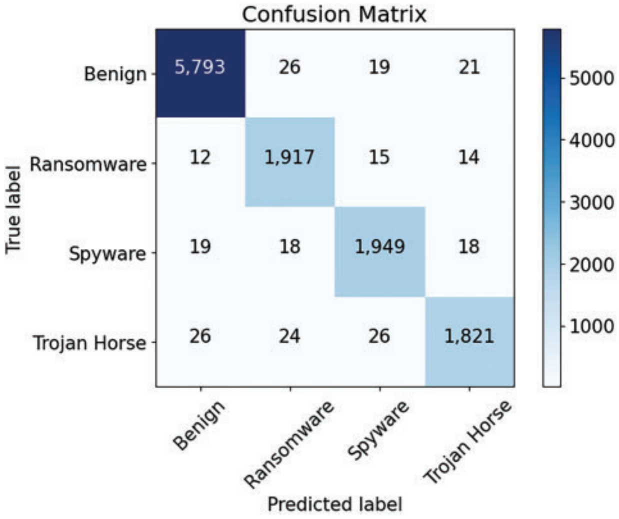

Figure 5: Multi-class classification confusion matrix

5 Evaluation and Comparison with Existing Approaches

Malware detection and classification is a challenging issue based on the dataset quality that needs to be solved.

The CIC-MalMem-2022 dataset has an obfuscated nature of the memory dumps observations that enable it to achieve high performance in binary classification problems while restricting the metrics values for the multiclass classification scenario. The contribution of this research was obtained via DNN-CNN model metrics values for the binary classification. This study delivers significant value through its revelation of preprocessing with data augmentation methods as well as the attention-based DNN model that outperforms convolutional neural networks when using CIC-MalMem-2022 data. Existing studies have carried out multiple experiments on various datasets with multiple categories of obfuscated malware, using a variety of features, several observations, and different techniques for data splitting and evaluation across applications. Multiple existing studies that used the attention layer gain accuracy less than our presented model accuracy. We used the attention layer mechanism to identify underlined patterns from the used data and hence were able to identify binary and multi classes more efficiently, as you can see in Table 5. We aim to make a fair comparison between our purpose model and those presented in significant previous works with similar techniques and goals, using a common test metric as the comparison criterion.

This measure includes accuracy for binary and multiclass classification. However, a direct comparison with similar works becomes challenging because each study approaches the privacy malware detection problem from its own unique perspective and situation while addressing different objectives for classification and handling obfuscated malware. Table 6 contains the comparison of this research, results with the other obfuscated malware classification contributions. In the case of a binary classifier, the accuracy score of 97.7% obtained in [13] using the Heterogeneous Deep Neural Network is less than 99.5% presented in this research.

Table 6: Comparison of this research results with other obfuscated malware classification contributions

Reference	Classification method	Classifier	Dataset	Metric	Values	Imbalance data
[5]	RandomForest (RF)	Binary	CIC-MalMem-2022	Accuracy	97.0%	×
[6]	Spatial attention and convolutional neural network (SACNN)	Multiclass	Malimg benchmark dataset	F1-score	97.32%	✓
[12]	Convolutional neural networks (CNN)	Multiclass	CIC-MalMem-2022	F1-score	72.0%	✓
[13]	Heterogeneous Deep Neuronal Network (HDNN)	Binary	360netlab DGA	Accuracy	97.7%	×
[19]	Tree-based Ensemble algorithms	Multiclass	CIC-MalMem-2022	Accuracy	96.0%	✓
[22]	Convolutional neural networks (CNN)	Multiclass	CIC-MalMem-2022	F1-score	75.0%	✓

(Continued)

Table 6 (continued)

Reference	Classification method	Classifier	Dataset	Metric	Values	Imbalance data
[24]	Decision Tree (DT)	Binary	CIC-MalMem-2022	Accuracy	98.0%	×
[26]	Random Forest (RF)	Multiclass	CIC-MalMem-2022	Accuracy	70.0%	✓
[17]	Attention-based multi-dimensional DL approach	Multiclass	IoMT cross-architecture benchmark dataset	Accuracy	94%	✓
[40]	One-Dimensional Convolutional Neural Network (1D-CNN)	Multiclass	CIC-MalMem-2022	Accuracy	88%	✓
[16]	Machine learning and big data approaches are used	Binary	CIC-MalMem-2022	Accuracy	98.41%	✓
[41]	BERT transformer-based model	Multiclass	CIC-MalMem-2022	Accuracy	74%	✓
Proposed method	**DNN-CNN**	**Binary** **Multiclass**	**CIC-MalMem-2022**	**Accuracy**	**99.5%** **97.9%**	× ✓

According to Table 6. in the comparison table, our proposed methodology achieved the highest accuracy score which is greater than reached by [13] equal to 97.7% without using parallel computing methods and using other malware classification techniques based on the 360netlab DGA dataset. We highly recommend that our presented attention-based approach be used on other datasets that tackle the binary obfuscated malware classification problem. In terms of multiclass classification, using the SMOTE technique to handle the imbalanced data to differentiate between obfuscated malware categories is insufficient on its own to obtain the highest metrics scores, but only by using this technique with the presented approach, it was possible.

6 Discussion

Our presented model of preprocessing methods delivers higher accuracy results than traditional Machine Learning methods since it addresses more complex multi-classification problems. Moreover, comparative analysis shows that the presented DNN-CNN obtains comparable and ever-best metrics to multiclass obfuscated malware classifiers developed using alternative detection methods. CIC-MalMem-2022 offers an appropriate memory dumping dataset to detect malicious samples against benign samples, enabling the development of robust classifiers. The CIC-MalMem-2022 dataset allows traditional deep learning methods to solve the binary classification task effectively. However, traditional machine learning algorithms face limitations when trying to detect and classify multiclass obfuscated malware through memory dump observations due to their complex patterns.

According to the above-mentioned comparison table, various classification methods and classifiers as well as datasets and performance metrics demonstrate multiple vital observations. Convolutional Neural Networks and Random Forests show that they perform outstanding on binary classification problems and achieve 97% to 98% accuracy. On the other hand, the 1D-CNN performs well on multiclass classification problems and achieves 88% accuracy on the CIC-MalMem-2022 dataset. Most research relies on accuracy as their main evaluation method while F1-score acts as a preferred metric for multiclass data to measure precision against the recall. Performance outcomes show substantial dataset dependence since the CIC-MalMem-2022 dataset appears in most analytical methods. The performance of algorithms depends on the type of data during evaluation processes because each dataset requires distinct assessment methods. Data balancing techniques substantially impact the operational performance of these classifiers. The performance of CNN-based models functions optimally under balanced data distribution conditions but competing models experience difficulty due to unbalanced classes which corrupt their achieved F1-scores. Additional data balancing approaches should be considered to enhance performance since current methods achieve only 59% accuracy on the target class. Furthermore, the analysis shows our proposed DNN-CNN model gained 99.5% accuracy on binary and 97.9% accuracy on multiclass. As per the above comparative analysis, it can be observed that our proposed model outperforms other approaches. The proposed systems can work with both binary, as well as multi-class attacks.

Due to this reason, deep learning is being utilized immensely along with novel data preprocessing techniques. In this study, we proposed a lightweight obfuscated malware detection attention-based DNN-CNN model for binary classification, such as benign and malware, and for multiple class detection including benign, trojan horse, ransomware, and spyware. We tested our proposed model on the CIC-MalMem-2022 dataset that tackles binary and multiclass problems. For binary classification, we gathered features through the Principal Component Analysis (PCA) technique, and for multiclass classification, we applied the SMOTE (data augmentation technique) to handle the imbalanced data.

7 Conclusion and Future Work

The CIC-MalMem-2022 malware memory dumping observations dataset provides an effective way to detect malicious samples from benign ones resulting in high-classifier performance metrics. For that corresponding purpose, traditional deep learning techniques can be utilized with the CIC-MalMem-2022 dataset to successfully handle the binary classification problem. However, traditional machine learning algorithm results are limited while detecting and classifying the multiclass obfuscated malware based on the memory dumping observation because of having complex patterns. Due to this reason, deep learning is being utilized immensely along with novel data preprocessing techniques. In this research, we proposed a lightweight obfuscated malware detection attention-based DNN-CNN model for binary classification, such as benign and malware, and for multiple class detection, including benign, trojan horse, ransomware, and spyware. We tested our proposed model on the CIC-MalMem-2022 dataset that tackles the binary and multiclass problems. For binary classification, we gathered features through the Principal Component Analysis (PCA) technique, and for multiclass classification, we applied the SMOTE technique (data augmentation technique) to handle the imbalanced data. In this research, we mainly focused on binary and multiclass classification by using novel techniques and reliance on a single dataset named CIC-MalMem-2022, which limited potential scalability, generalizability, and issues when applied to larger datasets. However, in the future, we will use our presented model for detecting sub-families of malware categories with the CIC-MalMem-2022 dataset through novel preprocessing techniques. We will also use the different datasets to evaluate the performance of our presented model.

Acknowledgement: The authors would like to express their deep thanks to the Vice Presidency for Scientific Research at Naif Arab University for Security Sciences for their kind encouragement of this work.

Funding Statement: This research was funded by Naif Arab University for Security Sciences under grant No. NAUSS-23-R11.

Author Contributions: Conceptualization, Emad Ul Haq Qazi and Muhammad Hamza Faheem; data curation, Waleed Khalid Al-Ghanem, Emad Ul Haq Qazi, Tanveer Zia, Muhammad Hamza Faheem, Muhammad Imran and Iftikhar Ahmad; formal analysis, Emad Ul Haq Qazi and Muhammad Hamza Faheem; funding acquisition, Emad Ul Haq Qazi; methodology, Emad Ul Haq Qazi and Muhammad Hamza Faheem; project administration, Emad Ul Haq Qazi; resources, Emad Ul Haq Qazi; software, Waleed Khalid AL-Ghanem, Emad Ul Haq Qazi, Tanveer Zia, Muhammad Hamza Faheem, Muhammad Imran and Iftikhar Ahmad; supervision, Emad Ul Haq Qazi; validation, Waleed Khalid AL-Ghanem, Emad Ul Haq Qazi, Tanveer Zia, Muhammad Hamza Faheem, Muhammad Imran and Iftikhar Ahmad; visualization, Waleed Khalid AL-Ghanem, Emad Ul Haq Qazi, Tanveer Zia, Muhammad Hamza Faheem, Muhammad Imran and Iftikhar Ahmad; writing—original draft, Emad Ul Haq Qazi and Muhammad Hamza Faheem; writing—review and editing, Waleed Khalid AL-Ghanem, Emad Ul Haq Qazi, Tanveer Zia, Muhammad Hamza Faheem, Muhammad Imran and Iftikhar Ahmad. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The data that support the findings of this study are available in the malware memory dataset CIC-MalMem-2022, https://www.unb.ca/cic/datasets/malmem-2022.html#:~:text=CIC%2DMalMem%2D2022&text= The%20dataset%20was%20created%20to,test%20obfuscated%20malware%20detection%20systems (accessed on 1 January 2025).

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. Vinayakumar R, Alazab M, Soman KP, Poornachandran P, Venkatraman S. Robust intelligent malware detection using deep learning. IEEE Access. 2019;7:46717–38. doi:10.1109/ACCESS.2019.2906934.
2. Jahromi AN, Hashemi S, Dehghantanha A, Parizi RM, Choo KKR. An enhanced stacked LSTM method with no random initialization for malware threat hunting in safety and time-critical systems. IEEE Trans Emerg Top Comput Intell. 2020;4(5):630–40. doi:10.1109/TETCI.2019.2910243.
3. Huseynov H, Kourai K, Saadawi T, Igbe O. Virtual machine introspection for anomaly-based keylogger detection. In: 2020 IEEE 21st International Conference on High Performance Switching and Routing (HPSR); 2020 May 11–14; Newark, NJ, USA. p. 1–6.
4. Amin M, Shehwar D, Ullah A, Guarda T, Tanveer TA, Anwar S. A deep learning system for health care IoT and smartphone malware detection. Neural Comput Appl. 2022;34:11283–94. doi:10.1007/s00521-020-05429-x.
5. Carrier T, Victor P, Teleoglu A, Lashkari AH. Detecting obfuscated malware using memory feature engineering. In: Proceedings of the 8th International Conference on Information Systems Security and Privacy (ICISSP 2022); 2022 Feb 9–11; Online. p. 177–88.
6. Awan MJ, Masood OA, Mohammed MA, Yasin A, Zain AM, Damaševičius R, et al. Image-based malware classification using VGG19 network and spatial convolutional attention. Electronics. 2021;10(19):2444. doi:10.3390/electronics10192444.
7. Alomari ES, Nuiaa RR, Alyasseri ZA, Mohammed A, Sani HJ, Esa NS, et al. Malware detection using deep learning and correlationbased feature selection. Symmetry. 2023;15(1):123. doi:10.3390/sym15010123.
8. Aljabri M, Alhaidari F, Albuainain A, Alrashidi S, Alansari J, Alqahtani W, et al. Ransomware detection based on machine learning using memory features. Egypt Inform J. 2024;25:100445. doi:10.1016/j.eij.2024.100445.

9. Chen CW, Su CH, Lee KW, Bair PH. Malware family classification using active learning by learning. In: 2020 22nd International Conference on Advanced Communication Technology (ICACT); 2020 Feb 16–19; Phoenix Park, Republic of Korea. p. 590–5.
10. Xu Y, Li D, Li Q, Xu S. Malware evasion attacks against IoT and other devices: an empirical study. Tsinghua Sci Technol. 2023;29(1):127–42. doi:10.26599/TST.2023.9010005.
11. Sana A, Muhammad A. Evaluation and classification of obfuscated Android malware through deep learning using ensemble voting mechanism. Sci Rep. 2023;13(1):3093. doi:10.1038/s41598-023-30028-w.
12. Shafin SS, Karmakar G, Mareels I. Obfuscated memory malware detection in re-source-constrained IoT devices for smart city applications. Sensors. 2023;23(11):5348. doi:10.3390/s23115348.
13. Yang L, Liu G, Dai Y, Wang J, Zhai J. Detecting stealthy domain generation algorithms using heterogeneous deep neural network frame-work. IEEE Access. 2020;8:82876–89. doi:10.1109/ACCESS.2020.2988877.
14. Setiawan H, Putro PAW, Pramadi YR. Comparison of lstm architecture for malware classification. In: 2020 International Conference on Informatics, Multimedia, Cyber and Information System (ICIMCIS); 2020 Nov 19–20; Jakarta, Indonesia. p. 93–7.
15. CIC. Malware memory analysis CIC-MalMem-2022 [Internet]. [cited 2024 Mar 11]. Available from: https://www.unb.ca/cic/datasets/malmem-2022.html.
16. Dener M, Ok G, Orman A. Malware detection using memory analysis data in big data environment. Appl Sci. 2022;12(17):8604. doi:10.3390/app12178604.
17. Ravi V, Pham TD, Alazab M. Attention-based multidimensional deep learning approach for cross-architecture IoMT malware detection and classification in healthcare cyber-physical systems. IEEE Trans Comput Soc Syst. 2022;10(4):1597–606. doi:10.1109/TCSS.2022.3198123.
18. Mishra P, Aggarwal P, Vidyarthi A, Singh P, Khan B, Alhelou HH, et al. VMShield: memory introspection-based malware detection to secure cloud-based services against stealthy attacks. IEEE Trans Ind Inform. 2021;17(10):6754–64. doi:10.1109/TII.2020.3048791.
19. Cletus A, Opoku A, Weyori B. A novel hybrid features with ensemble and data augmentation for efficient and resilient malware variant detection. Int J Eng Trends Technol. 2023;71:439–57. doi:10.14445/22315381/IJETT-V71I8P238.
20. Özkan K, Işık Ş., Kartal Y. Evaluation of convolutional neural network features for malware de-tection. In: 2018 6th International Symposium on Digital Forensic and Security (ISDFS); 2018 Mar 22–25; Antalya, Turkey. p. 1–5.
21. Akhtar MS, Feng T. Detection of malware by deep learning as CNN-LSTM machine learning techniques in real time. Symmetry. 2022;14(11):2308. doi:10.3390/sym14112308.
22. Smith D, Khorsandroo S, Roy K. Supervised and unsupervised learning techniques utilizing malware datasets. In: 2023 IEEE 2nd International Conference on AI in Cybersecurity (ICAIC); 2023 Feb 7–9; Houston, TX, USA. Vol. 2023, p. 1–7.
23. Malani H, Bhat A, Palriwala S, Aditya J, Chaturvedi A. A unique approach to malware detection using deep convolutional neural networks. In: 2022 4th International Conference on Electrical, Control and Instrumentation Engineering (ICECIE); 2022 Nov 26; Kuala Lumpur, Malaysia. p. 1–6.
24. Abualhaj MM, Al-Khatib SN. Using decision tree classifier to detect Trojan Horse based on memory data. Telkomnika. 2024;22(2):393–400. doi:10.12928/telkomnika.v22i2.25753.
25. Mezina A, Burget R. Obfuscated malware detection using dilated convolutional network. In: 2022 14th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT); 2022 Oct 11–13; Valencia, Spain. p. 110–5.
26. Mishra A, Bagade P. MalDicom: a memory forensic framework for detecting malicious payload in DICOM files. arXiv:2312.00483. 2023.
27. Mohamed N, Belaton B. SBI model for the detection of advanced persistent threat based on strange behavior of using credential dumping technique. IEEE Access. 2021;9:42919–32. doi:10.1109/ACCESS.2021.3066289.
28. Jones KJ, Wang Y. Malgazer: an automated malware classifier with running window entropy and machine learning. In: 2020 Sixth International Conference on Mobile and Secure Services (MobiSecServ); 2020 Feb 22–23; Gainesville, FL, USA. p. 1–6.

29. Yang J, Guo Y. AEFETA: encrypted traffic classification framework based on self-learning of feature. In: 2021 6th International Conference on Intelligent Computing and Signal Processing (ICSP); 2021 Apr 9–11; Xi'an, China. p. 876–80.

30. Mani G, Pasumarti V, Bhargava B, Vora FT, MacDonald J, King J, et al. DeCrypto Pro: deep learning based cryptomining malware detection using performance counters. In: 2020 IEEE International Conference on Autonomic Computing and Self-Organizing Systems (ACSOS); 2020 Aug 17–21; Washington, DC, USA. p. 109–18.

31. Vinayakumar R, Soman KP, Poornachandran P, Akarsh S, Elhoseny M. Deep learning framework for cyber threat situational awareness based on email and url data analysis. In: Hassanien AE, Elhoseny M, editors. Cybersecurity and secure information systems: challenges and solutions in smart environments. Berlin/Heidelberg, Germany: Springer; 2019. p. 87–124.

32. Hemalatha J, Roseline SA, Geetha S, Kadry S, Damaševičius R. An efficient densenet-based deep learning model for malware detection. Entropy. 2021;23(3):344. doi:10.3390/e23030344.

33. Tekerek A, Yapici MM. A novel malware classification and augmentation model based on convolutional neural network. Comput Secur. 2022;112:102515. doi:10.1016/j.cose.2021.102515.

34. Sihwail R, Omar K, Arifin KAZ. An effective memory analysis for malware De-tection and classification. Comput Mater Contin. 2021;67(2):2301–20. doi:10.32604/cmc.2021.014510.

35. Homayoun S, Dehghantanha A, Ahmadzadeh M, Hashemi S, Khayami R. Know abnormal, find evil: frequent pattern mining for ransomware threat hunting and intelligence. IEEE Trans Emerg Top Comput. 2017;8(2):341–51. doi:10.1109/TETC.2017.2756908.

36. Xiao G, Li J, Chen Y, Li K. MalFCS: an effective malware classification framework with automated feature extraction based on deep convolutional neural networks. J Parallel Distrib Comput. 2020;141:49–58. doi:10.1016/j.jpdc.2020.03.012.

37. Stiawan D, Daely SM, Heryanto A, Afifah N, Idris MY, Budiarto R. Ransomware detection based on opcode behavior using k-nearest neighbors algorithm. Inf Technol Control. 2021;50(3):495–506. doi:10.5755/j01.itc.50.3.25816.

38. Ghazi MR, Raghava NS. A scalable and stacked ensemble approach to improve intrusion detection in clouds. Inf Technol Control. 2023;52(4):898–914. doi:10.5755/j01.itc.52.4.32042.

39. Cevallos-Salas D, Grijalva F, Estrada-Jiménez J, Benítez D, Andrade R. Obfuscated privacy malware classifiers based on memory dumping analysis. IEEE Access. 2024;12:17481–98. doi:10.1109/ACCESS.2024.3358840.

40. Mahdi RH, Trabelsi H. Effective obfuscated malware detection leveraging cutting-edge machine and deep learning approaches. Int J Intell Eng Syst. 2021;18(1):1045.

41. Mashrur Arifin M, Suyehara Tolman T, Yeh JH. Unveiling the efficacy of BERT's attention in memory obfuscated malware detection. In: International Conference on Information Security Practice and Experience; 2024 Oct 25–27; Wuhan, China. Singapore: Springer Nature. p. 273–91.