Exploring the Application of Transfer Learning in Malware Detection by Fine-tuning Pre-Trained Models on Binary Classification to New Datasets on Multi-class Classification

1st Bamidele Ajayi
School of Computer Science
University of Sunderland
Sunderland, United Kingdom
0000-0003-1419-9375

2nd Basel Barakat
School of Computer Science
University of Sunderland
Sunderland, United Kingdom
0000-0001-9126-7613

3rd Ken McGarry
School of Computer Science
University of Sunderland
Sunderland, United Kingdom
0000-0002-9329-9835

4th Mays Abukeshek
School of Computer Science
University of Sunderland
Sunderland, United Kingdom
0009-0000-6340-0253

***Abstract*—This research presents a method for classifying malicious and benign binary files using Convolutional Neural Networks (CNNs), transitioning from binary to multiclass classification. Three commonly used datasets were tested: EMBER, BODMAS, and MALIMG, with EMBER and BODMAS serving as training and testing sets for the base model. Data from these datasets is converted into image representations and analyzed by CNN models, achieving a high accuracy of 98%.**

A transfer learning model is then developed, incorporating knowledge from EMBER and BODMAS. This model reduces training time significantly and achieves 97% accuracy with just 5 epochs and a batch size of 25 across 25 malware family sets, averaging a perfect AUC of 1.00. This indicates perfect discrimination between positive and negative classes, with 100% correct predictions, underscoring the robustness of the method.

***Index Terms*—Convolutional Neural Networks, Malware Detection, Transfer Learning, Cybersecurity, Ember Dataset, BODMAS Dataset, MALIMG Dataset.**

I. Introduction

The ever-changing landscape in technology has led to a significant rise in cyber threats, underscoring the urgent need for efficient methods to detect and classify malware. Traditional approaches often struggle to keep pace with the diverse and evolving nature of these threats. This research delves into leveraging Convolutional Neural Networks (CNNs) for transfer learning in classification tasks, specifically focusing on the nuanced distinction among various classes of malware.

Motivated by the limitations of binary classification in capturing the complexity of modern malware variants, this research transitions to multi-class classification. This approach allows for a more detailed and granular analysis, essential for robust cybersecurity measures. We utilize three pivotal datasets—EMBER, BODMAS, and MALIMG—employing EMBER and BODMAS for training and testing our model. These datasets are transformed into image representations and subjected to CNN models, achieving a remarkable level of accuracy.

Building upon the insights gained from our base model, we develop a transfer learning framework that not only drastically reduces training time but also achieves an outstanding accuracy rate of 97% after just 5 epochs. This framework demonstrates the model's capability to adapt and improve performance by leveraging previously acquired features when confronted with new datasets.

The methodology of this research critically examines transfer learning by fine-tuning a base model on a new dataset, illustrating how pre-existing knowledge can enhance model effectiveness in diverse classification tasks. Comparative analysis between the tuned model and the original model provides a comprehensive evaluation of transfer learning's efficacy in this context.

This research contributes significantly to the field of machine learning by showcasing how transfer learning can significantly improve accuracy and efficiency in classification tasks, potentially leading to more effective and adaptive AI systems in various applications with emphasis on sophisticated detection in the face of evolving cyber threats.

II. Related Work

Wang et al. [1] proposed a malware classification method based on transfer learning for multi-channel image vision features and ResNet convolutional neural networks, which can better extract the texture features of malware, effectively improve the accuracy and detection efficiency. A new framework utilizes transfer learning for visual classification of multi-channel malware, enhancing detection efficiency and accuracy, achieving 99.99% accuracy on the Microsoft BIG benchmark dataset.

Priya et al. [2], transfer learning was used for zero-day malware detection, where malware binaries are turned into grayscale images before being processed using models for classification based on transfer learning. The paper explores using transfer learning with models like AlexNet, VGG16, VGG19, GoogLeNet, and ResNet for malware classification by converting malware binaries into grayscale images.

In [3], a malware detection method based on transfer learning was proposed, where they use the pre-trained deep convolutional-based AlexNet architecture having ImageNet weights for feature extraction. The proposed transfer learning-based method effectively classifies malware into their families. The performance of the suggested model is compared to other contemporary ImageNet models.

The authors of [4], compared the performance of various machine learning and deep learning technologies towards malware classification such as Logistic Regression (LR), Artificial Neural Networks (ANN), Convolutional Neural Network (CNN), transfer learning on CNN and Long Short Term Memory (LSTM). Transfer learning using InceptionV3 achieved high accuracy (98.76% test, 99.6% train) for malware classification, outperforming LSTM and other models in the research.

The authors of [5], proposed a novel ensemble model, Stacked Ensemble (SE-AGM), composed of three light-weight neural network models (autoencoder, GRU, and MLP) for malware detection. Transfer learning was utilized for malware detection in IoT using a stacked ensemble model trained on essential features extracted from the MalImg dataset, achieving a high accuracy of 99.43

In [6], a new method based on Markov image and transfer learning on machine learning was proposed for malware detection and classification, and an experience comparing the performance of the proposed and grayscale methods was done. The paper proposes a method using Markov image and transfer learning for malware detection and classification, achieving high accuracy (0.973) and low loss (0.076), showing suitability for classification tasks.

AlGarni et al. [7] Efficient Convolutional Neural Network with Transfer Learning is utilized for malware classification, achieving a high accuracy of 99.93% by classifying malware families using pre-trained models. The role of deep convolution neural networks in malware classification and solutions for utilizing machine learning to detect and classify malware families through transfer learning are discussed.

Wang et al. [8] as mentioned in this paper implemented several EffiicientNet models into two types of Malware BIG 2015 that had been visualized into grayscale and RGB format, they found that EfficientNetB7 implemented into RGB dataset got 99.63% of accuracy, 98.36% of precision, 99.835% of recall, 98.34% of F1-score, and 98.30% of AUC, with only takes 10 epochs in the training process.EfficientNet, a transfer learning model, achieved 99.63% accuracy in malware classification using RGB datasets, outperforming other models with only 10 training epochs.

III. Methodology

The first step in developing the base model is to acquire the necessary data. This involves obtaining the training dataset from EMBER [10] and storing it in a specific directory. The dataset consists of 800,000 samples, each described by feature vectors and matching labels. Any entries without labels are removed from the dataset. Similarly, the test dataset, comprising 134,435 samples from BODMAS [11], is obtained from a specific directory, with any unlabeled entries also being removed from this dataset if relevant as shown in Figure 1

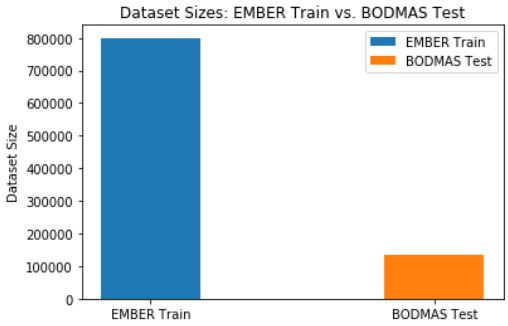

Fig. 1. EMBER BODMAS Dataset

After collecting the data, it goes through a procedure called pre-processing. This involves splitting the training dataset into two subsets: the training subset and the validation subset. The split is done according to a preset ratio. The data is normalized using the 'StandardScaler'. This method ensures that all features have a mean of 0 and a standard deviation of 1. Furthermore, the data is transformed to match the input specifications of the Conv2D layer in the CNN model.

The next step is creating the model architecture using the 'Keras' API. This design includes 'Conv2D' layers for extracting features, 'BatchNormalization' layers for normalizing, and 'Dense' layers for classification. In order to address the issue of overfitting, the technique of L2 regularization is employed. The hyperparameters of the model, such as the learning rate, number of epochs, and batch size, are also specified.

The model is subsequently constructed using the 'Adam optimizer', employing the sparse categorical crossentropy loss function, and evaluating its performance based on accuracy. The model is then trained using the training data, while the validation data is used to monitor the model's performance and prevent overfitting.

After the training process, the model produces predictions on the test data. To assess the model's performance, various performance measures such as the ROC curve, AUC, and confusion matrix(see Figure 5) are calculated.

The methodology also utilizes transfer learning by employing the pre-trained model as a base for additional training on a new dataset [18], as shown in Figure 2

This dataset consists of 7459 samples, with 25 different classes of malware families. The dataset is divided into 5221 training samples and 2238 test samples as shown in Figure 2. The dataset is effectively loaded and preprocessed using the ImageDataGenerator class from Keras. This includes resizing the images to a uniform size of 64×64 pixels and normalizing the data to ensure consistency in the input features. The preprocessing stage is crucial in preparing the dataset for further analysis and training of models.

After obtaining and preparing the dataset, an exploratory data analysis is performed. This stage is important for un-

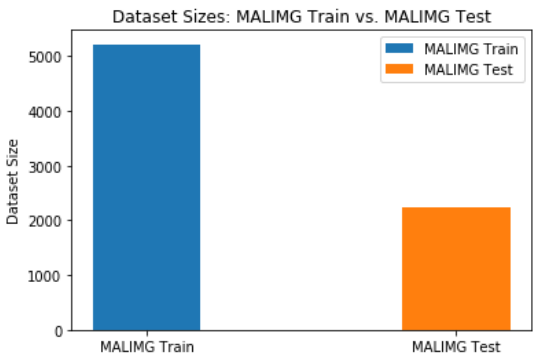

Fig. 2. MALIMG Dataset

derstanding the distribution of classes within the dataset and visually examining a subset of the malwares as shown in Figure 3

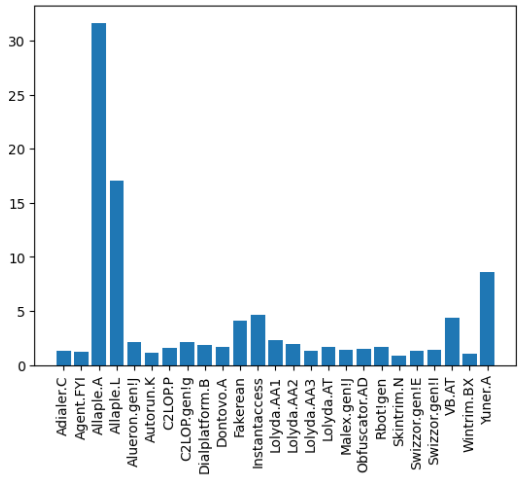

Fig. 3. MALIMG Malware Distribution

TABLE I
BASE MODEL PERFORMANCE METRICS FOR THE MODEL

Metric	Value
Train Loss	0.1082
Train Accuracy	0.9726
Validation Loss	0.2452
Validation Accuracy	0.9310
Test Loss	0.1143
Test Accuracy	0.9790

Transfer learning is employed by adapting a pre-trained model to the new task. This process involves modifying the initial and final layers and incorporating two new layers to suit the specific requirements of the task. Subsequently, the model is trained on the new dataset and assessed using comparable metrics to those employed for the original model.

The training performance, ROC curve, and confusion matrix are graphically shown to offer a deeper understanding of the

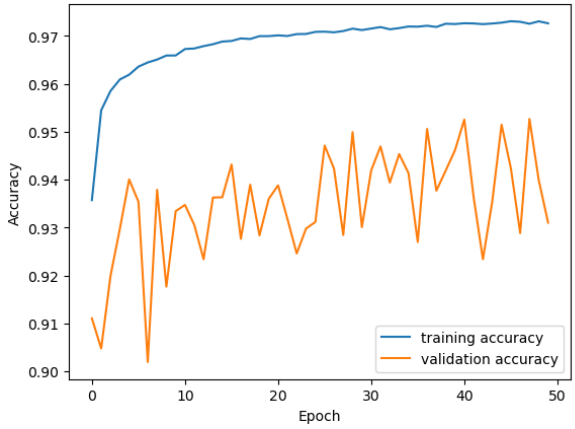

Fig. 4. Base Model Performance Graph

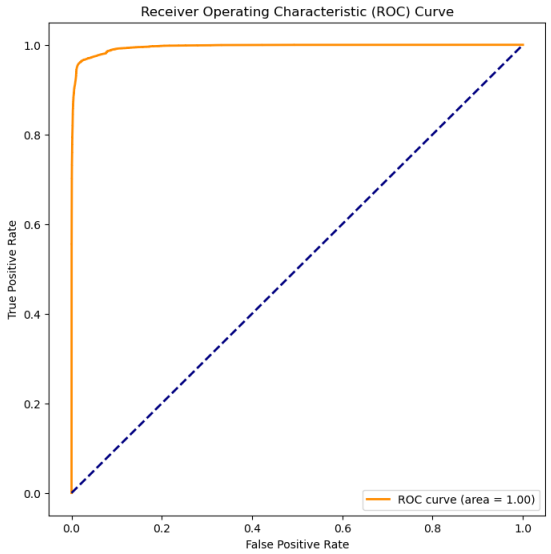

Fig. 5. Base Model ROC

model's learning progress and effectiveness.

IV. IMPLEMENTATION

Our transfer learning approach for malware classification incorporates data from both EMBER and BODMAS. The training dataset comprises 800,000 samples. The test dataset consists of around 134,435 samples which were used to build the base model.

Once the data is retrieved, it is partitioned into three distinct sets: training, validation, and test. A substantial proportion of the data is assigned to the validation set in order to assess the model's performance throughout training. The input features are normalized using the StandardScaler function from the 'sklearn.preprocessing' library. This is done to ensure that all features are scaled uniformly, which is crucial for the optimal performance of neural network models. The data is converted to align with the specifications of Conv2D layers. Transformed into a 2D tensor to facilitate convolutional processes.

The fundamental architecture consists of two Conv2D layers with ReLU activation functions, each of which is subsequently followed by BatchNormalization to retrieve information. A Flatten layer transforms 2D feature maps into a 1D vector. The base model consists of several interconnected layers using rectified linear unit (ReLU) activation functions. Batch normalization is applied after each layer, except for the last layer, which utilizes a softmax activation function for binary classification. L2 regularization is implemented in all layers, excluding the input layer, to mitigate overfitting. The base model is compiled using the Adam optimizer, sparse categorical crossentropy loss function, and accuracy as the evaluation metric. The training process consists of multiple epochs, where each epoch involves processing a batch of data from both the training and validation datasets. The model defines hyperparameters such as learning rate, batch size, and batch size. As an illustration, the learning rate is set to 0.001, the number of epochs is set to 50, and the batch size is set to 1000 based on manual adjustments during training. Our base model shows consistent improvement in both training and validation accuracies suggesting that the model is learning relevant patterns from the data without significant overfitting as shown in Figure 4.

In order to assess the model's performance, we examine the test dataset to determine metrics such as accuracy, loss, and other pertinent measures. The Receiver Operating Characteristic (ROC) curve and Area Under the Curve (AUC) are employed to evaluate the model's ability to differentiate across classes. The high AUC values indicate better discrimination between classes. The smooth ROC curve across training and validation sets suggests good generalization ability of the base model, as shown in Figure 5.

During our research's subsequent stage, transfer learning is utilized by modifying the base model to fulfill the needs of the MALIMG task as shown in Figure 6. This process entails the removal of layers from both the beginning and end of the base model and the incorporation of layers each for the respective ones previously removed that are specifically designed for MALIMG.

The transfer learning model is improved by incorporating additional layers, such as enabling the model to function in inference mode and providing a layer for multi-class classification. Adjusting the updated model entails utilizing the same Adam optimizer, employing the categorical crossentropy loss function, and evaluating correctness. Model checkpoints are utilized to store the model according to its validation accuracy. During the training, it was necessary to adjust the hyperparameters, such as the learning rate, epochs, and batch size. As an illustration, in the context of our transfer learning, we opted to leave the learning rate to 0.001 as is from the base model(pre-trained model), training the transfer learning model for 5 epochs, and utilizing a batch size of 32 employing a callback function to store the model according to the validation accuracy.

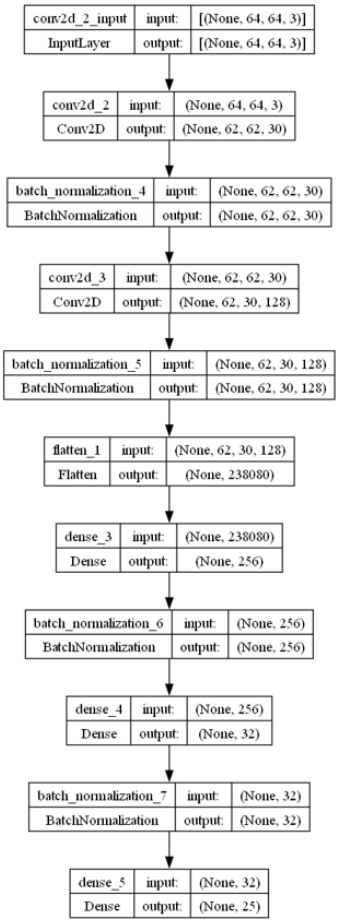

Fig. 6. Transfer Learning Model Architecture

V. Results and Evaluation

The transfer learning model demonstrates high performance, across training, validation and test datasets consistently achieving over 96% accuracy as shown in Figure 8. This indicates performance on both unfamiliar and unseen data. The loss values are relatively low suggesting a good fit with room for improvement. The consistency between validation and test metrics indicating minimal overfitting and solid generalization capabilities.

A detailed breakdown of classification metrics reveals precision, recall and F1 scores of 1.00 for classes such as 0, 1, 2, 5, 8, 9, 11, 12, 14, 16, 17, 18, and 24. Class 3 exhibits near perfect metrics with a decrease in recall (0.99) resulting in an F1 score of 0.99. Class 4 shows a minor decline in recall (0.96) with an F1 score of 0.98 See Table IV. Classes 6 and 7 display considerably lower scores; Class 6 has a precision of 0.76, a recall of 0.51 and an F1 score of 0.61; Class 7 has a precision of 0.86, a recall of 0.91 and an F1 score of 0.88. Classes 10, 13 and 15 experience slight reductions, in either precision or recall, while maintain F1 scores ranging from (97–99), as shown in Table IV.

In classes 19 and 23 there is a decrease, in effectiveness with F1 scores of 0.97 and 0.89 because of reduced recall. Classes

20 and 21 have the lowest metrics with class 20 having a precision of 0.50 recall of 0.66 and F1 score of 0.57 while Class 21 has a precision of 0.43 recall of 0.49 and F1 score of 0.46 See Table IV.

The overall accuracy score of 0.97 indicates performance which is generally high. The average precision across all categories is at a level of 0.94 along with recall and an F1 score at 0.93 treat all classes equally irrespective of support. The model appears to perform well for most classes achieving perfect precision, recall and F1 scores for many classes' except for the following; six (6) seven (7) twenty (20) and twenty-one (21) which show notably lower performances indicating challenges faced by the model in handling these specific classes possibly due to data imbalance or class complexity See Table IV.

To improve the performance in these classes data augmentation methods can be utilized to increase sample sizes for those classes while implementing class specific enhancements, like targeted feature extraction or custom loss functions could aid in addressing difficulties posed by these challenging classes. Adjusting the hyperparameters precisely and implementing regularization methods could potentially lower the loss values.

The micro average ROC curve demonstrates an Area Under the Curve (AUC) value of 1.0 showing how effectively our model can distinguish between classes of malware family see Figure 9.

TABLE II
COMPARISON OF THE ACHIEVED ACCURACY AND EPOCHS WITH THE LITERATURE

Model	Accuracy (%)	Epochs	Dataset
EfficientNetB7	99.63	10	Microsoft BIG [8]
LSTM	99	500	MALIMG [14]
VGG16	88.40	30	MALIMG [16]
Custom model	98.7	25	MALIMG [16]
LSTM	99	321	MALIMG [17]
LSTM	94	30	Microsoft BIG [17]
CNN	97	5	MALIMG (Ours)

TABLE III
TRANSFER LEARNING MODEL PERFORMANCE METRICS

Metric	Value
Transfer Learning Train Loss	1.4020
Transfer Learning Train Accuracy	0.9753
Transfer Learning Validation Loss	1.3052
Transfer Learning Validation Accuracy	0.9692
Transfer Learning Test Loss	1.3052
Transfer Learning Test Accuracy	0.9692

Our transfer learning model as it exhibited exceptional performance enhancements leveraging optimized hyperparameters from the base model, namely by reducing the number of epochs and setting the batch size to 5 and 32 respectively, we were able to get impressive results in considerably shorter periods compared to other works with higher epochs [8], [14]–[17].

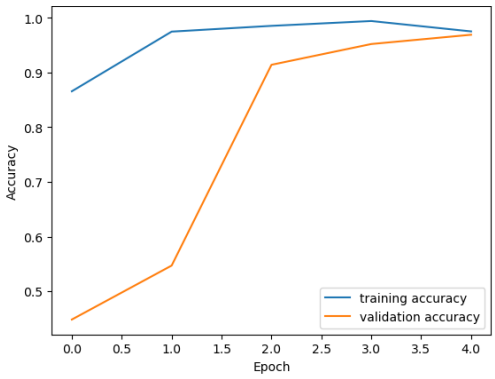

Fig. 7. Transfer Learning Model Performance Graph

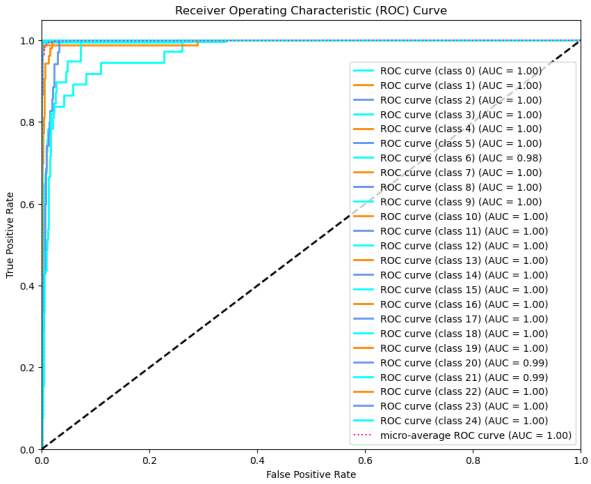

Fig. 8. Transfer learning Model ROC

Moreover, by using transfer learning and without careful hyperparameter tuning, we have significantly shown our model's performance making it more adaptable, to changing threat landscapes with fewer epochs and performance implications.

VI. CONCLUSION

This research contributes significantly to advancing the field of cybersecurity by demonstrating the effectiveness of transfer learning combined with tailored hyperparameters in enhancing the efficiency and accuracy of malware classification models. By refining and adapting a foundational binary classification model for multi-class classification using transfer learning, we underscore the adaptability and scalability of this approach. We demonstrated how existing neural network architectures can be optimized and repurposed through transfer learning to tackle complex and diverse malware family classification tasks. This adaptation not only improves classification accuracy but also reduces the computational resources and time required for model training, making it a practical solution for real-world cybersecurity challenges.

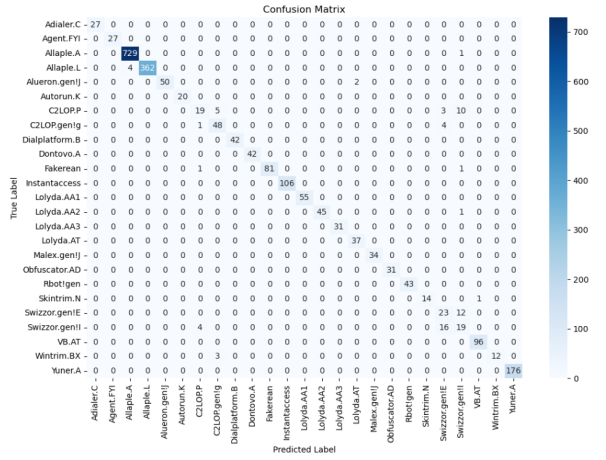

Fig. 9. Transfer learning Model Confusion Matrix

TABLE IV
CLASSIFICATION REPORT METRICS

Class	Precision	Recall	F1-score	Support
0	1.00	1.00	1.00	27
1	1.00	1.00	1.00	27
2	0.99	1.00	1.00	730
3	1.00	0.99	0.99	366
4	1.00	0.96	0.98	52
5	1.00	1.00	1.00	20
6	0.76	0.51	0.61	37
7	0.86	0.91	0.88	53
8	1.00	1.00	1.00	42
9	1.00	1.00	1.00	42
10	1.00	0.98	0.99	83
11	1.00	1.00	1.00	106
12	1.00	1.00	1.00	55
13	1.00	0.98	0.99	46
14	1.00	1.00	1.00	31
15	0.95	1.00	0.97	37
16	1.00	1.00	1.00	34
17	1.00	1.00	1.00	31
18	1.00	1.00	1.00	43
19	1.00	0.93	0.97	15
20	0.50	0.66	0.57	35
21	0.43	0.49	0.46	39
22	0.99	1.00	0.99	96
23	1.00	0.80	0.89	15
24	1.00	1.00	1.00	176
Accuracy			0.97	2238
Macro avg	0.94	0.93	0.93	2238
Weighted avg	0.97	0.97	0.97	2238

REFERENCES

[1] Zhao, Z.; Yang, S.; Zhao, D. A New Framework for Visual Classification of Multi-Channel Malware Based on Transfer Learning. Appl. Sci. 2023, 13, 2484. https://doi.org/10.3390/app13042484

[2] Priya, V., and Sathya Sofia, A. *Review on Malware Classification and Malware Detection Using Transfer Learning Approach*. In *2023 5th International Conference on Smart Systems and Inventive Technology (ICSSIT)*, pp. 1042-1049, 2023.

[3] Chakraborty, A., Kumar, S. (2023). Transfer Learning-Based Malware Classification. In: Thakur, M., Agnihotri, S., Rajpurohit, B.S., Pant, M., Deep, K., Nagar, A.K. (eds) Soft Computing for Problem Solving. Lecture Notes in Networks and Systems, vol 547. Springer, Singapore. https://doi.org/10.1007/978-981-19-6525-8_3

[4] Ahmed, M., Afreen, N., Ahmed, M., Sameer, M., and Ahamed, J. *An inception V3 approach for malware classification using machine learning and transfer learning*. *International Journal of Intelligent Networks*, 2023, 11-18.

[5] Panda, P., CU, O. K., Marappan, S., Ma, S., S, M., and Veesani Nandi, D. *Transfer learning for image-based malware detection for IoT*. *Sensors*, 2023, 3253.

[6] Kwan, L. M. (2022, November). Markov Image with Transfer Learning for Malware Detection and Classification. In TENCON 2022-2022 IEEE Region 10 Conference (TENCON) (pp. 1-6). IEEE.

[7] AlGarni, M. D., AlRoobaea, R., Almotiri, J., Ullah, S. S., Hussain, S., and Umar, F. *An efficient convolutional neural network with transfer learning for malware classification*. *Wireless Communications and Mobile Computing*, 2022, pp. 1-8.

[8] Pratama, H. Y., and Sidabutar, J. *Malware classification and visualization using EfficientNet and B2IMG algorithm*. In *2022 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, pp. 75-80, 2022. IEEE.

[9] Anderson, H.S., and Roth, P. *Ember: an open dataset for training static PE malware machine learning models*. arXiv preprint arXiv:1804.04637, 2018.

[10] Anderson, H.S. and Roth, P., 2018. Ember: an open dataset for training static pe malware machine learning models. arXiv preprint arXiv:1804.04637.

[11] Yang, Limin, Ciptadi, Arridhana, Laziuk, Ihar, Ahmadzadeh, Ali, and Wang, Gang. *BODMAS: An Open Dataset for Learning based Temporal Analysis of PE Malware*. In *4th Deep Learning and Security Workshop*, 2021.

[12] Naeem, Muhammad Rehan. *MalImg dataset.zip*. figshare, 2023. https://doi.org/10.6084/m9.figshare.24189882.v1

[13] Musaad Darwish AlGarni, Roobaea AlRoobaea, Jasem Almotiri, Syed Sajid Ullah, Saddam Hussain, and Fazlullah Umar. *An Efficient Convolutional Neural Network with Transfer Learning for Malware Classification*. *Wireless Communications and Mobile Computing*, vol. 2022, Article ID 4841741, 8 pages, 2022. https://doi.org/10.1155/2022/4841741

[14] Panda, Pratyush, Om Kumar C U, Suguna Marappan, Suresh Ma, Manimurugan S, and Deeksha Veesani Nandi. *Transfer Learning for Image-Based Malware Detection for IoT*. *Sensors*, vol. 23, no. 6, 2023, pp. 3253. https://doi.org/10.3390/s23063253

[15] P. Aggarwal, S. F. Ahamed, S. Shetty and L. J. Freeman, "Selective Targeted Transfer Learning for Malware Classification," 2021 Third IEEE International Conference on Trust, Privacy and Security in Intelligent Systems and Applications (TPS-ISA), Atlanta, GA, USA, 2021, pp. 114-120, doi: 10.1109/TPSISA52974.2021.00013.

[16] Dipendra Pant and Rabindra Bista. *Image-based Malware Classification using Deep Convolutional Neural Network and Transfer Learning*. In *2021 3rd International Conference on Advanced Information Science and System (AISS 2021)*, pp. 9, 2021, Sanya, China. ACM, New York, NY, USA. https://doi.org/10.1145/3503047.3503081

[17] Marastoni, N., Giacobazzi, R., and Dalla Preda, M. *Data augmentation and transfer learning to classify malware images in a deep learning context*. *Journal of Computer Virology and Hacking Technology*

[18] B. Barakat and Q. Huang, "Enhancing Transfer Learning Reliability via Block-Wise Fine-Tuning," 2023 International Conference on Machine Learning and Applications (ICMLA), Jacksonville, FL, USA, 2023, pp. 414-421, doi: 10.1109/ICMLA58977.2023.00064.