International Journal of Mathematical, Engineering and Management Sciences
Vol. 10, No. 4, 913-930, 2025
https://doi.org/10.33889/IJMEMS.2025.10.4.044

Ram Arti Publishers

Efficient Malware Classification using Transfer Learning and Stacked Ensemble Techniques

Krishna Kumar
Department of Information Technology, College of Technology,
G. B. Pant University of Agriculture and Technology, Pantnagar, Uttarakhand, India.
Corresponding author: 57871@gbpuat.ac.in, krishna.kumar20011@gmail.com

Hardwari Lal Mandoria
Department of Information Technology, College of Technology,
G. B. Pant University of Agriculture and Technology, Pantnagar, Uttarakhand, India.
E-mail: drmandoria11@gmail.com

Rajeev Singh
Department of Computer Engineering, College of Technology,
G. B. Pant University of Agriculture and Technology, Pantnagar, Uttarakhand, India.
E-mail: rajeevpec@gmail.com

(Received September 17, 2024; Revised on December 12, 2024; Accepted on January 30, 2025)

Abstract
The exponential growth of internet usage and communication devices has led to heightened security vulnerabilities, including the proliferation of malware such as viruses, ransomware, trojans, and spyware. These increasingly sophisticated malware variants pose significant challenges in their detection and classification. The existing visualization-based deep learning approach addresses some of these challenges but often requires extensive computational resources and prolonged training times and is prone to overfitting. This work proposed a transfer learning-based stacked ensemble technique to enhance the efficiency and accuracy of a malware classification model. The six scaled variants of the EfficientNetB0 architecture are selected for their performance on the ImageNet dataset. Their scalability was trained on the Malimg dataset, which comprises 9,339 malware images across 25 categories. Leveraging transfer learning for feature extraction significantly reduced training time and achieved a competitive accuracy of 99.10% within fewer epochs. To further enhance performance, the study employed a stacked ensemble approach by combining the strengths of three high-performing transfer learning models into two ensemble configurations: an average ensemble and a weighted average ensemble. The weighted average ensemble model demonstrated superior performance, achieving a remarkable training accuracy of 99.84% and a validation accuracy of 99.25%. These results underscore the effectiveness of the proposed approach in addressing modern malware classification challenges efficiently.

Keywords- Transfer learning, Malware classification, Convolutional neural network, Ensemble learning, Cyber security.

1. Introduction

In the modern digital era, exponential growth is seen in using smart electronic devices connected to the internet. The data generated from the electronic devices and controlling signals transfers through the network and generates the data for business intelligence purposes (Kumar and Dwivedi, 2017; Dwivedi et al., 2018). Every device connected to the modern communication network is not completely free from the risk of cybersecurity threats (Chauhan et al., 2021). There has been a 72% rise in data breaches since 2021, with an average data breach cost of $4.45 million. There are 35% of malware delivered via email, and the number of ransomware victims rose by 128.17% between 2022 and 2023 (John and Swanston, 2024). According to security researchers there is a 30% year-on-year increase in global cyberattacks (22nd July-Threat Intelligence Report, 2024). There is continuous research work required to defend against the challenging cybersecurity threats occurring due to distributed workforces (such as working from home), the expansion

of IoT devices and cloud services, and the rapid increase in ransomware attacks. AI-powered smart IoT devices play a crucial role in building sustainable public health surveillance systems, managing smart cities effectively (Ahad et al., 2023), and providing early solutions and predictions for geohazards, Industry 4.0, and data-driven business analytics (Rane et al., 2024). With humans being surrounded by digital devices connected to the internet, the increasing statistical data on cybersecurity attacks drives researchers and cybersecurity professionals to enhance cybersecurity models using the latest advanced technologies.

The use of AI and ML techniques works fine with traditional network intrusion detection systems, where the network traffic data is processed (Rai and Mandoria, 2019; Pujari et al., 2022; Chauhan et al., 2021; Husain et al., 2010). This approach is not much more effective when the data packets are encrypted and obfuscated. The encrypted malicious data packets that are undetected by traditional network intrusion detection systems might be ensembled at the destination source as malware. The various known and unknown variants of viruses, worms, Trojan horses, logic bombs, ransomware, advanced persistent attacks, and zero-day attacks are used with different attack vectors (Rai and Mandoria, 2019). The attack vector is the path used by the attacker to gain unauthorized access to a system, network, or application.

The traditional signature-based and behavior-based intrusion detection systems were bypassed by the novel and unknown, encrypted, and obfuscated malware. The recent literature inferred that the visualization-based malware detection and classification approach is providing better results for the various variants of known and unknown malware. The malware visualization approach converts the malware executable code into greyscale images by making groups of 8 binary digits in the range of 0-to-255-pixel values of the image (Nataraj et al., 2011). This makes use of computer vision techniques like convolutional neural networks and transfer learning techniques. The transfer learning approach uses learning experience to solve the new related problems that make the task easier and faster. The use of transfer learning eliminates the efforts to learn feature extraction (Rodríguez et al., 2022; Jung et al., 2021).

The recent research focuses on optimization in terms of accuracy, efficiency, and resource utilization. The vision-based malware detection and classification employs the potential of computer vision. The computer vision-based techniques require a large image dataset for the deep learning models, which requires high computational resources and time to train the model. The transfer learning technique is effective in reducing the extra computational resources and time needed to train the model.

The main contribution of the paper is the following:
1) To analyze the recent research work in area of intrusion detection and malware classification.
2) Performance analysis of transfer learning-based EfficientNet models.
3) Performance optimization using the ensemble approach by selecting the best-performing base models.

The remaining components of the paper are given as follows: Section 2 describes the related works; Section 3 presents the proposed research methodology used in experimental work. The experimental evaluation is done in Section 4. The Section 5 describes the conclusion and future scope of the work.

2. Related Works

The research work done in recent years in image-based malware detection and classification is discussed in this section. The visualization or image-based malware classification primarily uses the CNN technology. Nataraj et al. (2011) proposed a classification model that was able to classify 25 malware families of the 9458 malware samples with a classification accuracy of 97.18%. They have used the GIST (Oliva and Torralba, 2001) to compute the texture feature and k-nearest neighbor with Euclidean distance for the classification, along with 10-fold cross-validation.

Vinayakumar et al. (2019) developed a two-stage intrusion detection model that detects malware and classifies it into malware classes. The one-dimensional CNN and LSTM (long short-term memory) based model achieves accuracy of 96.3% and 98.8% on grayscale image dataset Malimg and a customized private dataset respectively.

Go et al. (2020) proposed the classical malware classification ResNeXt50 that achieved the accuracy of 98.32% and 98.86% on the original Malimg dataset and the modified Malimg dataset respectively. The model is trained on the two-dimensional grayscale malware images of size 224 × 224, which requires higher computational resources and training time.

Aslan and Yilmaz (2021) introduced a novel DL-based architecture to classify different malware variations and families based on a combination of two pre-trained transfer learning models. The model uses visualization techniques in which the binary code of the malware is converted to a fixed-sized grayscale image. The resized image is passed as an input to the proposed CNN model, which combines pre-trained models to extract features. The model achieves classification accuracy of 94.88%, 96.5%, and 97.78% on the Microsoft Big 2015, Malevis, and Malimg datasets, respectively.

Awan et al. (2021) proposed an enhanced CNN based model that incorporates dynamic spatial convolution and VGG19 for feature selection and freezing base layers. The model was trained on the Malimg dataset, which has 25 types of malware samples in image files. The model is able to classify malware samples with accuracy of 97.68%.

El-Sayed et al. (2021) proposed seven visualization-based classification algorithms and the performance of the models are evaluated on the basis of computational cost and accuracy. The PCAP files are converted into a colored image to learn the image structure and patterns. The VGG16 and SVM achieve an accuracy of 96% and 94%, respectively, among all classifiers. This model for classifying malware based on images proves valuable in identifying known and unknown network intrusions. Moreover, it provides opportunities for future improvements in accuracy.

O'Shaughnessy and Sheridan (2022) proposed a visualization-based approach to minimize the issues faced by the existing approach to malware classification. Space-filling curve technique is used for visual feature extraction on a dataset of obfuscated and non-obfuscated 13,599 samples of malware from 23 families. The space-filling curve is a method used to map between 1D space and 2D space, from the binary executable file. The Binary executable file is a sequence of bytes as a 1D array converted into a 2D grayscale image. The proposed approach begins the work with malware conversion to image samples then feature extraction task is followed by the malware classification. It achieves the classification accuracy of 97.6%.

Lu et al. (2022) proposed a self-attentive based real-time malware classification model. It is an ensemble of Vision Transformer and CNN to enhance accuracy and reduce the inference latency of the model that provides the 98.17% accuracy. The vision transformer technique requires more computing resources and training time as compared to the proposed approach.

Seneviratne et al. (2022) use the newer technique of Vision Transformer (ViT) (Dosovitskiy et al., 2020), a self-supervised DL model for malware detection. It is an alternative and scalable approach for processing sample images. The first step is to split an image into patches (such as 8x8 or 16x16 pixels). The standard transfer encoder embeds these patches with additional information to create an original image with reduced feature images. The model SHERLOCK (Seneviratne et al., 2022) achieves the detection accuracy of 97% with precision value of 87% on the dataset MalNet (Freitas et al., 2022).

Zhong et al. (2023) introduced a new classification technique having three modules. A malware binary is transformed into an image via the converter. The feature engineer uses a contrast-limited adaptive histogram equalization technique to raise the local contrast in different image regions and then resizes the treated image to a smaller, fixed size to speed up the classification process. The classifier uses a shallow CNN-based model with 96% classification accuracy.

Ahmed et al. (2023) proposed an InceptionV3 and transfer learning approach in which the four ML models (LR, ANN, CNN, and LSTM) and the transfer learning model InceptionV3 are trained on the 9 classes of malware dataset MicrosoftBig2015 up to 100 epochs. In this work, the InceptionV3 model achieves the highest test accuracy of 98.76%.

Al-Qadasi et al. (2024) introduced advanced models, ConvNeXtV1 and V2, which achieved impressive classification accuracies of 99.47% and 98.91% on the large-scale image datasets Malimg and MaleVis, respectively. However, despite their high accuracy, these models, as part of the conventional CNN approach, demand higher computational cost and training time.

Vasan et al. (2024) proposed a GPU-free and feature engineering-free model, IMCBL, as an alternative to deep learning to provide a fast and lightweight model. The model is trained up to 50 epochs on the five different malware datasets. The model takes a total of 822 seconds to train using the CPU and achieves a classification accuracy of 97.64% on the Malimg dataset.

Habib et al. (2024) evaluated the six deep learning models InceptionV3, CNN, VGG16, ResNet50, MobileNet, and EfficientNetB0. In this experimental work, all the models are trained and tested on the Malimg dataset. The EfficientNetB0 model gives a higher classification accuracy of 99.14%, maintaining the balance of accuracy and resource utilization. However, the models are individually trained without using the transfer learning and ensemble learning approaches and achieve higher accuracy, but in general, deep learning models such as ResNet50 and VGG16 require lots of computational power and training time for the large image dataset.

Salas and de Geus (2024) proposed a transfer learning-based model, MobileNetFT, that is a fine-tuned MobileNet model. The proposed model is trained and tested on the three malware datasets (Microsoft Big 2015, Malimg, and Malevis) and a fusion dataset (combining all the dataset samples). The model achieves the highest accuracy of 99.07% on the Malimg dataset. However, the model does not achieve comparable accuracy on the other datasets, but by using the transfer learning approach, the training time is significantly reduced.

Puneeth et al. (2024) proposed the CNN model RMDNet, which includes the three-stage complex architecture of two parallel convolutional layers for binary and multiclass malware classification. The model is trained and tested on the RGB binary class dataset, the Malimg dataset, and the custom dataset dumpware 10. The model achieves an accuracy of 99.15%, 99.26%, and 98.19%, respectively, on these datasets. However, the model achieves higher classification accuracy on the Malimg dataset, but the proposed model demands high-performing computational resources and training time.

The related works indicate that while several deep learning (DL) models have been proposed for visualization-based malware classification, they often demand significant computational resources and extended training times. Despite this, these models still face challenges regarding accuracy and efficiency. These limitations can be addressed by incorporating transfer learning and ensemble techniques. Transfer learning allows models pre-trained on large image datasets to perform feature extraction efficiently on

similar types of images. Additionally, the ensemble approach can enhance accuracy by combining multiple base models, as individual models often struggle with overfitting issues.

3. Research Methodology

The proposed methodology includes the use of a real malware image dataset along with performance enhancement using transfer learning and ensemble learning approaches. The transfer learning based pre-trained model reduces the training time and resources for feature extraction; as a substitute, it reuses pre-trained values of weights and bias (**Figure 1**).

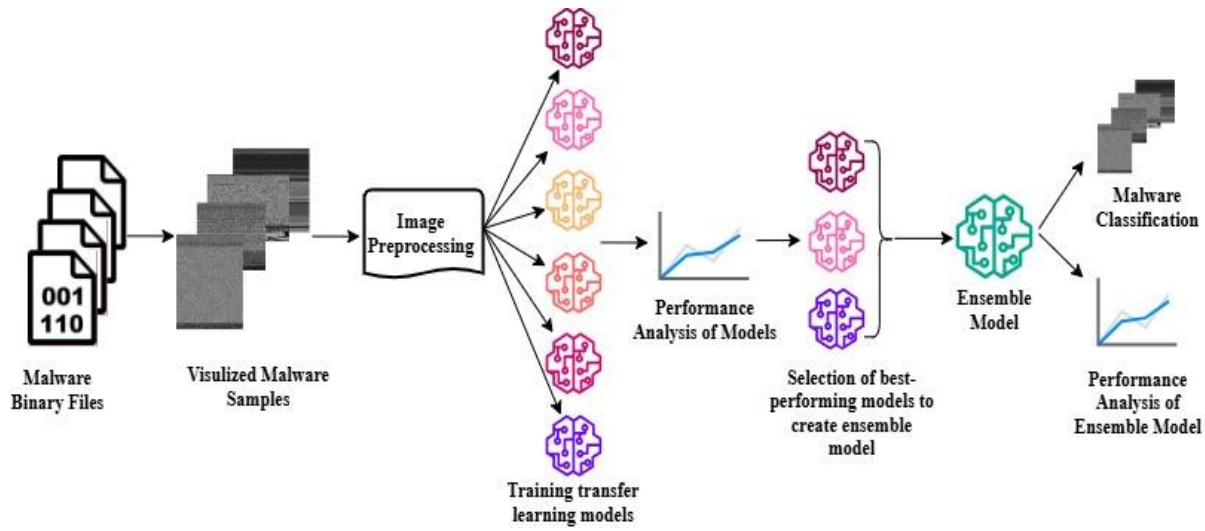

Figure 1. Proposed methodology for malware classification.

3.1 Hardware and Software Requirement

The experimental work is performed on the Google Collaboratory platform, which provides the required libraries. It provides high computing resources such as an A100 GPU with higher storage memory. This virtual computing has installed Python (version 3.10.12), TensorFlow (version 2.15.0), and the Numpy library.

3.2 Benchmark Dataset and Pre-processing

The models are trained and tested using the malware image dataset Malimg (Nataraj et al., 2011). It is the most prominent malware image dataset, with many researchers using it to train and test malware classification models. The malware picture dataset consists of 9339 grayscale images representing 25 malware classifications. The dataset is uneven, with the malware class *Allapale.A* having the most 2949 samples and the malware class *Skintrim.N* having the fewest 80 image samples (as shown in **Table 1**).

Table 1. Malware classes, type, and sample distribution of benchmark dataset.

S. No.	Malware classes	Malware type	Number of samples
1.	Adiler.C	Dialer	122
2.	Agent.FYI	Backdoor	116
3.	Allaple.A	Worm	2,949
4.	Allaple.L	Worm	1,591
5.	Alueron.gen!J	Trojan	198
6.	Autorun.K	Worm	106
7.	C2LOP.P	Trojan	200
8.	C2LOP.gen!g	Trojan	146
9.	Dialplatform.B	Dialer	177
10.	Dontovo.A	Downloader	162
11.	Fakerean	Rogue	381
12.	Instantaccess	Dialer	431
13.	Lolyda.AA1	Password stealing	213
14.	Lolyda.AA2	Password stealing	184
15.	Lolyda.AA3	Password stealing	123
16.	Lolyda.AT	Password stealing	159
17.	Malex.gen!J	Trojan	136
18.	Obfuscator.AD	Downloader	142
19.	Rbot!gen	Backdoor	158
20.	Skintrim.N	Trojan	80
21.	Swizzor.gen!E	Downloader	128
22.	Swizzor.gen!I	Downloader	132
23.	VB.AT	Worm	408
24.	Wintrim.BX	Downloader	97
25.	Yuner.A	Worm	800
Total			**9339**

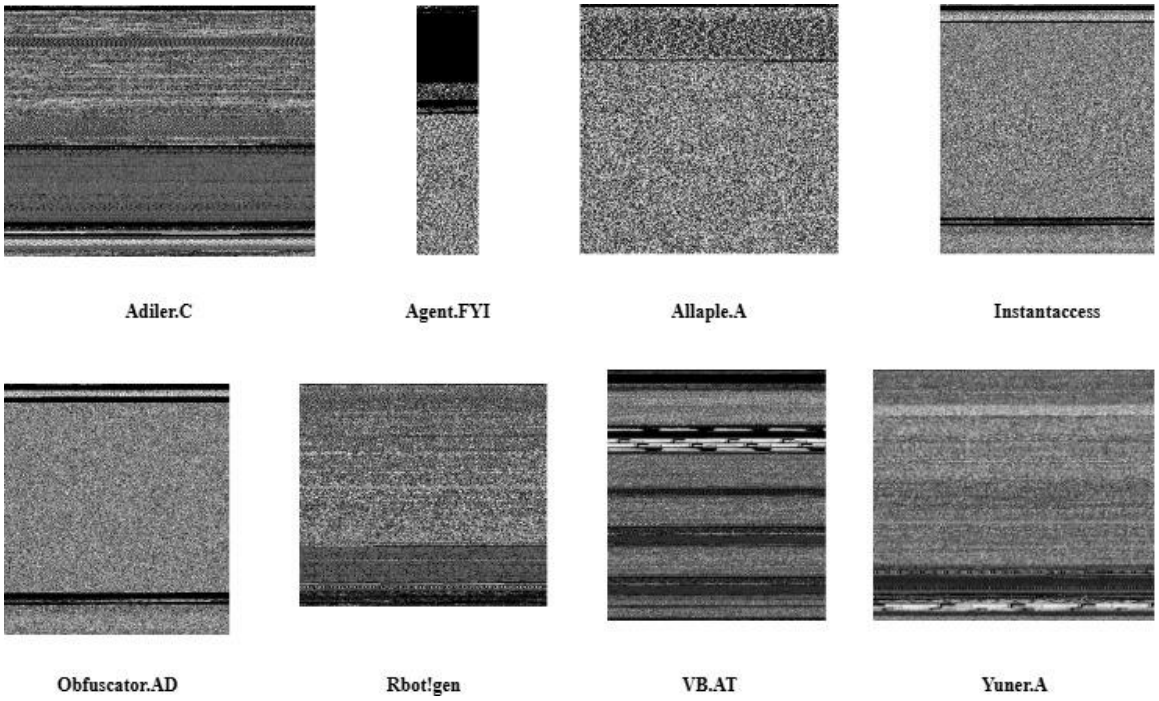

Figure 2. Greyscale image samples of Malimg dataset (Nataraj et al., 2011).

The visualized grayscale malware image samples are shown in **Figure 2**. The original Malimg dataset contains malware images having different sizes. Therefore, the image samples are resized to 224×224 pixels, and all pixels are normalized in the range from 0 to 1. The dataset is divided into a ratio of 80:20 for the training and testing.

3.3 Transfer Learning Approach

Transfer learning is a powerful technique for enhancing malware detection and classification systems. Research has shown that utilizing transfer learning in malware detection models can significantly improve accuracy and efficiency (Chen and Cao, 2023; Jung et al., 2021; Rodríguez et al., 2022). The transfer learning model uses the saved weight values learned from the 'ImageNet' dataset of 1000 classes. The selected models, like EfficientNetB3 and EfficientNetB7, provide an acceptable trade-off between accuracy and efficiency as compared to other models like ResNet-50, InceptionV2, and ResNet-152 (Tan and Le, 2019).

This experimental work uses the six high-performing scaled versions of base model EfficientNetB0 as shown in **Table 2**. These models are selected on the basis of higher accuracy and minimum model size. The conventional CNN model becomes more complex and tends to overfit on a large number of trainable parameters.

Table 2. Details of the transfer learning models used for malware classification.

S. No.	Models	Year	Size	Trainable parameters	Non-trainable parameters	Total parameters
1.	EfficientNetB3	2019	48	799769	10783535	11,583,304
2.	EfficientNetB7	2019	256	1324057	64097687	65,421,744
3.	EfficientNetV2B0	2021	29	668697	5919312	65,88,009
4.	EfficientNetV2S	2021	88	668697	20331360	21,000,057
5.	EfficientNetV2M	2021	220	668697	53150388	53,819,085
6.	EfficientNetV2L	2021	479	668697	117746848	118,415,545

3.4 Ensemble Learning Approach

Ensemble learning is an effective machine learning technique that combines multiple base models to create a more accurate and robust predictive model. It is used for both machine learning (ML) and deep learning (DL) models, combining heterogeneous or homogeneous models to optimize performance. Researchers are frequently using this technique in diverse fields, like stock price predictions (Majiid et al., 2023), healthcare (Krishna and Kokil, 2024), braille character recognition (Elaraby et al., 2024), and remote sensing applications (Fayaz et al., 2024).

The examples of ensemble approaches that work with homogeneous models are bagging, random forests, boosting, extreme gradient boosting, AdaBoost and gradient boosting. The ensemble approach that uses multiple heterogeneous models is called the stacked ensemble method (Mohammed and Kora, 2023). In this paper, the multiple heterogeneous transfer learning models are used as a base model in parallel to create a separate high-performing metamodel for more accurate prediction.

By aggregating the predictions of several base learners, ensemble methods can reduce the variance and bias and improve the overall performance compared to individual models. The ensemble model is created by selecting the best-performing transfer learning models. In the proposed approach, the ensemble technique is also used along with the transfer learning technique. In our case, there are heterogeneous transfer learning models for which the stacked ensemble approach is more suitable and applied. Therefore, the stacked ensemble method is used in two different ways, as discussed below.

1) *Average Ensemble Model*: The average ensemble method uses the outputs of selected base models by taking average of their predictions. For N models, the formula is:

$$\hat{y} = \frac{1}{N}\sum_{i=0}^{N} \hat{y}_i \quad (1)$$

where, $\hat{y}$ is the final average prediction, N is the number of models, and $\hat{y}_i$ is the prediction made by the i-th model.

2) *Weighted Average Ensemble Model*: The weighted average ensemble method assigns a weight to each base model's prediction based on its importance or accuracy. The final prediction is the weighted sum of the individual base model predictions. For *N* models, the formula is:

$$\hat{y} = \frac{1}{N}\sum_{i=0}^{N} w_i . \hat{y}_i \quad (2)$$

where, w_i is the weight assigned to the i-th base model's prediction, and $\sum_{i=1}^{N} w_i = 1$.

By applying the average ensemble method, we give equal preference to all models, but in a weighted ensemble model, we can prioritize the preference for a certain model over another.

3.5 Hyperparameters

In convolutional neural networks (CNNs), hyperparameters play a critical role in defining the architecture and training process. These hyperparameters can be broadly categorized into two groups: *Architectural hyperparameters* and *Training hyperparameters*. The architectural hyperparameters are pre-defined in the models, as in our case of the transfer learning approach, like the number of filters (kernels), filter size, typically 3×3, stride, number of layers, and pooling type and size. The top layers for pre-trained models are customized by adding the classification layers using the *ReLU* activation functions in the internal layer and *Softmax* in the final layer. We fine-tuned the training hyperparameters to customize the model to get the best results. The training hyperparameters are given in **Table 3**.

Table 3. Training hyperparameter used in experimental work.

S. No.	Hyperparameters	Value	Description
1.	Learning rate	0.001	Controls the step size during the optimization process.
2.	Batch size	32	Samples are divided into batches and processed before model updated.
3.	Epochs	30	Number of iterations or passes on training dataset.
4.	Optimizer	Adam	Dictates how the model weights are updated.
5.	Loss function	Categorical crossentropy	Used to find the variations on the actual and predicted output.

4. Experimental Evaluations and Results

This section describes the results obtained from the experimental work. The performances of transfer learning models and ensemble models are evaluated on the benchmark dataset. The original dataset contains malware images with varied dimensions.

Therefore, the malware images are resized to a 224×224-pixel image size as required for transfer learning models. In transfer learning models, only the top layers of pre-trained models are customized and trained on the targeted dataset.

4.1 Performance Analysis of Transfer Learning Models

The EfficientNetB0 is the base model that is designed for high-resolution image classification. This study utilizes the upgraded versions of EfficientNetB0, including EfficientNetB3, EfficientNetB7,

EfficientNetV2B0, EfficientNetV2S, EfficientNetV2M, and EfficientNetV2L. These six selected transfer learning models are trained on a malware dataset of 7,460 samples up to 30 epochs. **Figure 3** illustrates the training accuracy and loss curves, as well as the validation accuracy and loss curves across the 30 epochs. Among all models, EfficientNetV2B0 achieves the highest performance.

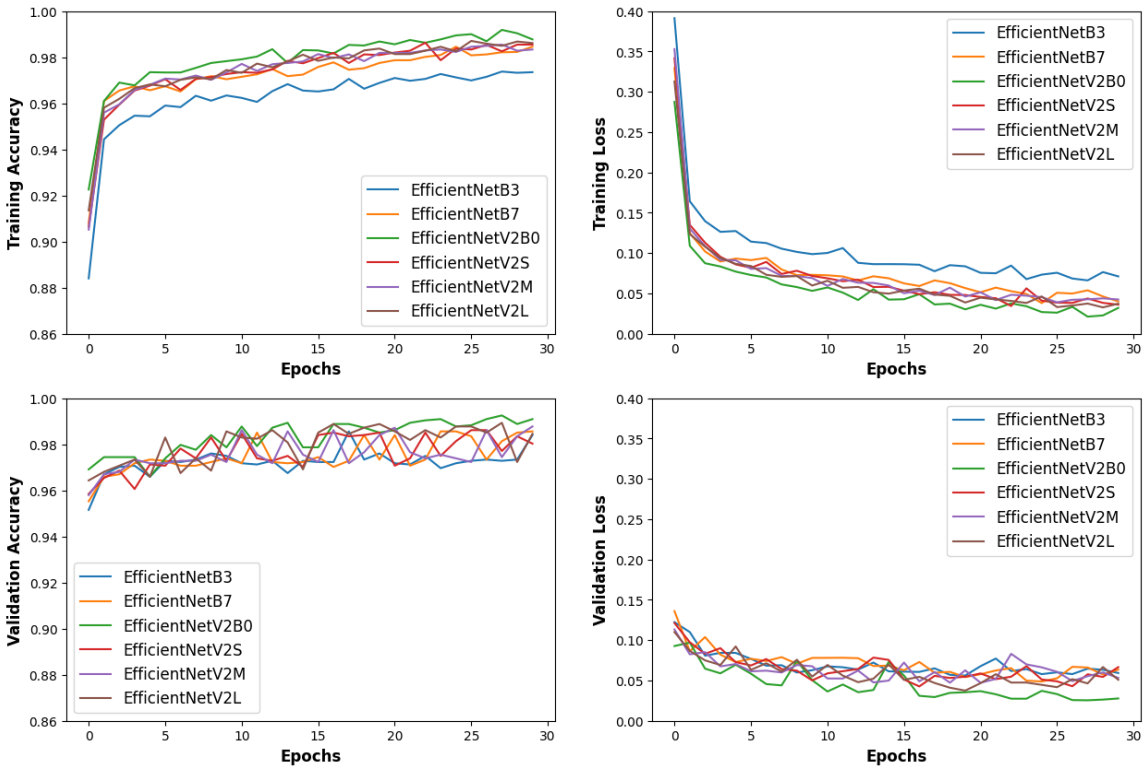

Figure 3. Accuracy and loss curve of transfer learning models.

The performance of all models is evaluated in terms of training time, training accuracy, and validation accuracy, as shown in **Table 4**. The EfficientNetV2B0 model achieves the highest training accuracy of 98.78% and validation accuracy of 99.10% among all models. Transfer learning-based deep learning models require less training time.

Table 4. Performance analysis of transfer learning models on benchmark dataset.

S. No.	Models	Training time (s)	Training accuracy (%)	Validation accuracy (%)
1.	EfficientNetB3	12 min 28 s	97.35	98.40
2.	EfficientNetB7	15 min 07 s	98.46	98.56
3.	EfficientNetV2B0	11 min 26 s	98.78	99.10
4.	EfficientNetV2S	8 min 45 s	98.55	98.03
5.	EfficientNetV2M	9 min 31 s	98.34	98.78
6.	EfficientNetV2L	14 min 09 s	98.62	98.46

All models except EfficientNetB7 take a training time of less than 15 minutes and achieve comparable accuracy in only 30 epochs. Therefore, by using the transfer learning approach, the training time is significantly reduced along with the computational cost.

The top-performing transfer learning models, EfficientNetV2B0, EfficientNetV2S, and EfficientNetV2M, are utilized to create a stacked ensemble model, aiming to further improve the model's accuracy and overall performance.

4.2 Performance Analysis of Ensemble Models

The stacked ensemble technique combines multiple trained models as base models to create a new meta-model. The base models can be incorporated into the meta-model in two ways, depending on their performance: the *average ensemble model* and the *weighted average ensemble model*.

In this study, two stacked ensemble models are developed using the trained transfer learning models EfficientNetB7, EfficientNetV2B0, and EfficientNetV2M. These ensemble models are further trained on the training dataset for 10 epochs. **Table 5** presents the performance of the ensemble models, including training time, training accuracy, and validation accuracy.

Table 5. Performance of ensemble models.

S. No.	Ensemble models	Base models	Training time (s)	Training accuracy (%)	Validation accuracy (%)
1.	Average ensemble model	EfficientNetB7, EfficientNetV2B0, and EfficientNetV2M	24 min 47 s	99.84	99.10
2.	Weighted average ensemble model	EfficientNetB7, EfficientNetV2B0, and EfficientNetV2M	31 min 11 s	99.84	99.25

Table 6. Performance matrix of average ensemble model.

Malware classes	Precision (%)	Recall (%)	F1-Score (%)	Samples
Adiler.C	100	100	100	32
Agent.FYI	100	100	100	15
Allaple.A	100	100	100	623
Allaple.L	100	100	100	326
Alueron.gen!J	100	100	100	45
Autorun.K	100	100	100	21
C2LOP.P	94	100	97	31
C2LOP.gen!g	100	98	99	42
Dialplatform.B	100	100	100	34
Dontovo.A	100	100	100	26
Fakerean	100	99	99	70
Instantaccess	100	100	100	75
Lolyda.AA1	98	100	99	44
Lolyda.AA2	100	96	98	25
Lolyda.AA3	100	100	100	25
Lolyda.AT	100	100	100	40
Malex.gen!J	100	100	100	27
Obfuscator.AD	100	100	100	25
Rbot!gen	100	100	100	35
Skintrim.N	100	100	100	20
Swizzor.gen!E	71	80	75	25
Swizzor.gen!I	72	59	65	22
VB.AT	100	100	100	84
Wintrim.BX	95	100	97	18
Yuner.A	100	100	100	149
Accuracy			**99**	**1879**
Macro avg	**97**	**97**	**97**	**1879**
Weighted avg	**99**	**99**	**99**	**1879**

The average ensemble model with three base models, EfficientNetB7, EfficientNetV2B0, and EfficientNetV2M, achieves a training accuracy of 99.84% and a validation accuracy of 99.10%. The weighted average ensemble model with similar base models performs well with a training accuracy of 99.84% and a validation accuracy of 99.25%.

The proposed methodology provides better results as compared to the individual model used by many researchers. The ensemble approach is more suitable with transfer learning to reduce the computational resources and training time required to train all the base models and the ensemble models.

Table 6 displays the performance metrics for the average ensemble model. The model achieves a precision value of 100% for 20 out of the 25 malware classes. However, the model faces challenges in distinguishing between the *Swizzor.gen!E* and *Swizzor.gen!I* class due to similarities in their malware image textures. The models are often confused with other similar classes, such as *Allaple.A*, *Allaple.L*, or *Lolyda.AA1*, *Lolyda.AA2*, and *Lolyda.AA3*. However, the model achieves remarkable performance for these classes.

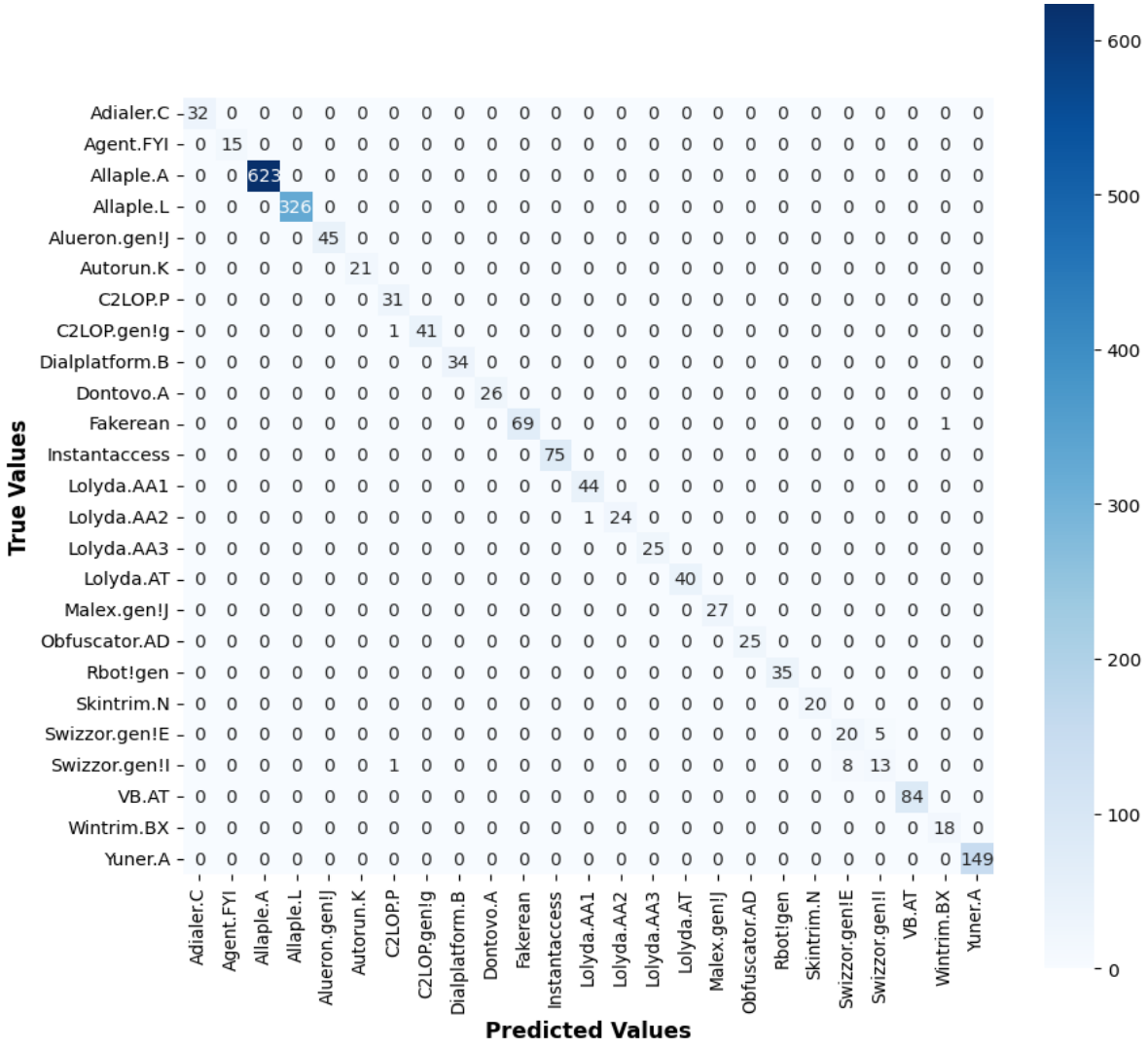

Figure 4. Confusion matrix for the average ensemble model.

Figure 4 presents the confusion matrix for the average ensemble model, highlighting its classification accuracy across all malware classes in the test dataset. This matrix shows class-wise performance of model by comparing actual class labels with predicted ones. The proposed average ensemble model correctly classified all samples in 20 out of the 25 classes in the test dataset. In the matrix, true values are displayed along the rows, while predicted values are shown along the columns.

The performance matrix provides only the calculated metrics, while the confusion matrix offers a detailed breakdown of the components used to derive those metrics. In the confusion matrix, correctly predicted values are represented along the diagonal, while misclassifications are displayed outside the diagonal. For instance, out of 25 *Lolyda.AA2* samples, 24 were accurately classified, while one sample was misclassified as *Lolyda.AA1*. This misclassification likely resulted from the similarity between the samples.

In the test dataset, the classes *Allaple.A* and *Allaple.L*, despite their identical naming, were correctly classified by the ensemble model with 100% accuracy. This demonstrates that the integration of transfer learning with ensemble techniques can be effectively utilized for a variety of multiclass classification problems. It is particularly well-suited for datasets containing diverse samples with minimal similarity between classes, enabling optimal performance.

Table 7. Performance matrix for weighted average ensemble model.

Malware classes	Precision (%)	Recall (%)	F1-Score (%)	Samples
Adiler.C	100	100	100	32
Agent.FYI	100	100	100	15
Allaple.A	100	100	100	623
Allaple.L	100	100	100	326
Alueron.gen!J	100	100	100	45
Autorun.K	100	100	100	21
C2LOP.P	91	100	95	31
C2LOP.gen!g	100	98	99	42
Dialplatform.B	100	100	100	34
Dontovo.A	100	100	100	26
Fakerean	100	99	99	70
Instantaccess	100	100	100	75
Lolyda.AA1	98	100	99	44
Lolyda.AA2	100	96	98	25
Lolyda.AA3	100	100	100	25
Lolyda.AT	100	100	100	40
Malex.gen!J	100	100	100	27
Obfuscator.AD	100	100	100	25
Rbot!gen	100	100	100	35
Skintrim.N	100	100	100	20
Swizzor.gen!E	79	88	83	25
Swizzor.gen!I	82	64	72	22
VB.AT	100	100	100	84
Wintrim.BX	95	100	97	18
Yuner.A	100	100	100	149
Accuracy			**99**	**1879**
Macro avg	**98**	**98**	**98**	**1879**
Weighted avg	**99**	**99**	**99**	**1879**

The Weighted Average Ensemble Model, combining the best-performing transfer learning models EfficientNetB7, EfficientNetV2B0, and EfficientNetV2M was trained on the training dataset and evaluated on the test dataset. This model demonstrated superior performance compared to the Average Ensemble Model. The performance metrics for the Weighted Average Ensemble Model are presented in **Table 7**.

Notably, the F1-scores for the classes *Swizzor.gen!E* and *Swizzor.gen!I* improved to 83% and 72%, respectively, highlighting the enhanced classification capability of the weighted ensemble approach.

Figure 5 illustrates the confusion matrix for the weighted average ensemble model. This model improves the classification accuracy for challenging classes in Malimg dataset like Swizzor.gen!E and Swizzor.gen!I. This model has improved accuracy and more correctly classifies the class Swizzor.gen!E as compared to the average ensemble model.

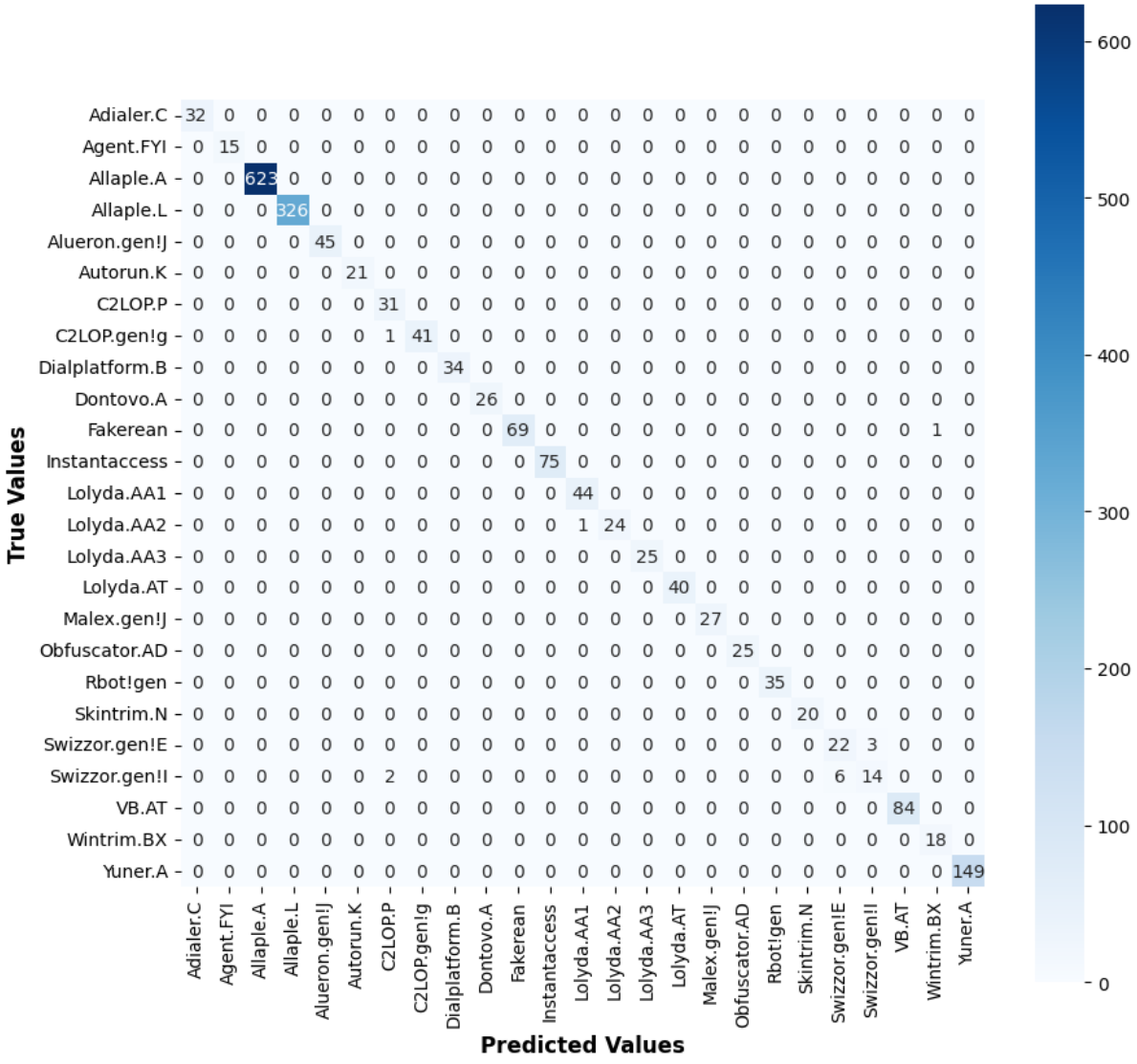

Figure 5. Confusion matrix for the weighted average ensemble model.

Figure 6 illustrates the Receiver Operating Characteristic (ROC) curves and corresponding Area Under the Curve (AUC) values for all evaluated transfer learning and ensemble models. The AUC values, ranging between 0 and 1, highlight the model's performance. The highest ROC-AUC value of 0.9886 is achieved by the weighted average ensemble model, while the lowest AUC value of 0.9674 is recorded for the

EfficientNetB3 model. The figure demonstrates that applying ensemble techniques significantly enhances the performance of classification models.

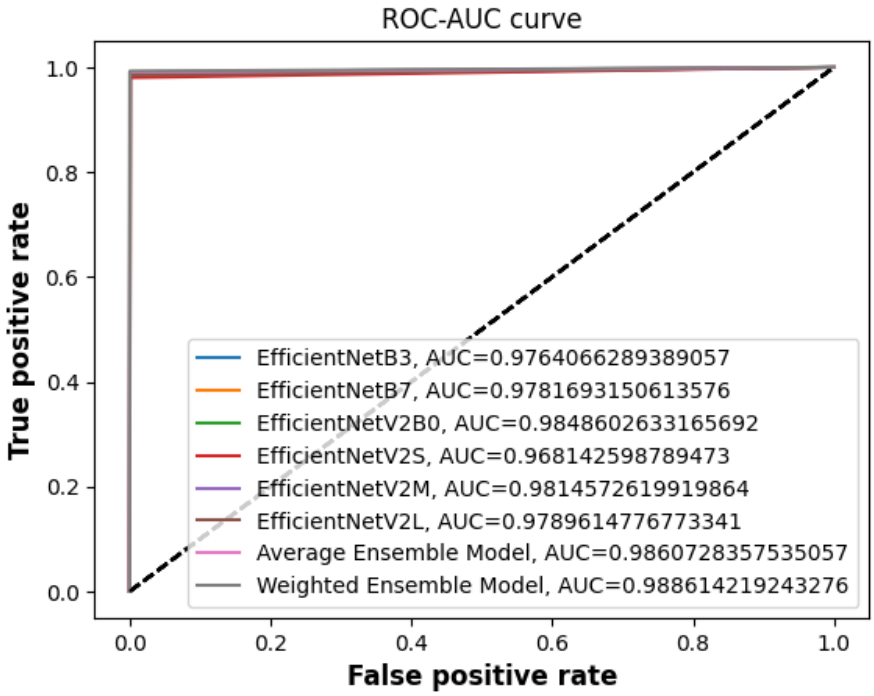

Figure 6. ROC-AUC curve for all models.

4.3 Comparison with Current State-of-The-Art

The comparison of results obtained from the proposed methodology with existing research studies is presented in **Table 8**. The proposed models are trained and evaluated on the benchmark dataset Malimg, and the performance is compared with existing literature works that used the same dataset. The proposed models achieve a comparable accuracy to recent works such as IMCBL (Vasan et al., 2024), EfficientNetB0 (Habib et al., 2024), and RMDNet (Puneeth et al., 2024).

The weighted average ensemble model demonstrates exceptional performance and outperformed the most existing models by achieving a test accuracy of 99.25%. The proposed transfer learning approach is significantly able to reduce resource requirements, computational costs, and training time. The ensemble technique further helps to enhance model accuracy to achieve optimized and efficient solutions.

The transfer learning-based EfficientNetB0 model achieves a notable validation accuracy of 99.10% within a training time of just 11 minutes and 26 seconds (as shown in **Table 4**). In the weighted average ensemble model, a higher weight is given to the EfficientNetB0 model compared to the other two base models due to its superior performance. The EfficientNetV2S model demonstrates remarkable efficiency by achieving a validation accuracy of 98.03% with the shortest training time of 8 minutes and 45 seconds, surpassing the performance of the IMCBL model (Vasan et al., 2024).

The transfer learning approach plays a crucial role in enhancing the model's efficiency. It significantly reduced the training time and minimized the computational resources during the feature extraction process

by focusing only on training the top layers for the classification task. Besides this, the weighted average ensemble method further improves the performance and classification accuracy up to 99.25%.

Table 8. Comparison of proposed model with current state-of-the-art.

References	Model architecture	Input image size	Accuracy (%)
Nataraj et al. (2011)	GIST, KNN	Original Images	97.18
Vinayakumar et al. (2019)	1D-CNN, LSTM	32×32	96.3
Go et al. (2020)	ResNeXt50	224×224	98.86
Aslan and Yilmaz (2021)	ResNet50, AlexNet, Inception-v3	224×224	97.78
Awan et al. (2021)	VGG19	224×224	97.68
Zhong et al. (2023)	VisMal	64×64	96
Al-Qadasi et al. (2024)	ConvNeXtV2	224×224	99.47
Vasan et al. (2024)	IMCBL	224×224	97.64
Habib et al. (2024)	EfficientNetB0	224×224	99.14
Salas and de Geus (2024)	MobileNetFT	224×224	99.07
Puneeth et al. (2024)	RMDNet	224×224	99.26
Proposed Model	**Average Ensemble Model**	**224×224**	**99.10**
Proposed Model	**Weighted Average Ensemble Model**	**224×224**	**99.25**

5. Conclusion and Future Scope

The experimental results demonstrate that transfer learning models are highly effective in reducing the computational cost associated with feature extraction in model training. The transfer learning models are pre-trained on the extensive ImageNet dataset, which contains over a million images across 1,000 classes. Transfer learning significantly reduces training time, simplifies model complexity, and mitigates the risk of overfitting in complex models. Pre-trained models are easy to use with minor modifications to the top layers, which makes them efficient for image-based classification tasks.

The performance of transfer learning models is further enhanced using the ensemble approach, which combines the strengths of top-performing base models. The transfer learning-based EfficientNetV2B0 model achieved a test accuracy of 99.10%, and the stacked ensemble approach improved this accuracy to 99.25% and outperformed many existing models. These results underscore the effectiveness and applicability of the proposed techniques.

Future research could explore the broader application of transfer learning and ensemble methods in information security. Additionally, addressing challenges such as dataset imbalance and limited sample sizes can be achieved using generative adversarial network (GAN) techniques, paving the way for tackling more complex problems in the domain.

Conflict of Interest

The authors confirm that there is no conflict of interest to declare for this publication.

AI Disclosure

During the preparation of this work the author(s) used generative AI in order to improve the language of the article. After using this tool/service, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the content of the publication.

Acknowledgments

I would like to thank my advisor for providing guidance, support, and encouragement throughout the entire work. Their mentorship and expertise were invaluable in helping me shape the direction and focus of my research work and bring in new ideas. I am also thankful to the Ministry of Social Justice & Empowerment, India, for providing financial support as an RGNF-SC fellowship. The authors would like to thank the editor and anonymous reviewers for their comments that helped improve the quality of this work.

References

22nd July - Threat Intelligence Report. (2024). Weekly intelligence report. Available: https://research.checkpoint.com/wp-content/uploads/2024/07/Threat_Intelligence_News_2022-07-22.pdf. Accessed in September 2024.

Ahad, M.A., Casalino, G., & Bhushan, B. (2023). *Enabling technologies for effective planning and management in sustainable smart cities*. Springer, Cham. ISBN: 978-3-031-22921-3(p), 978-3-031-22922-0(e). https://doi.org/10.1007/978-3-031-22922-0.

Ahmed, M., Afreen, N., Ahmed, M., Sameer, M., & Ahamed, J. (2023). An inception V3 approach for malware classification using machine learning and transfer learning. *International Journal of Intelligent Networks, 4*, 11-18. https://doi.org/10.1016/j.ijin.2022.11.005.

Al-Qadasi, H., Benchadi, D.Y.M., Chehida, S., Fukui, K., & Bensalem, S. (2024). Neural network innovations in image-based malware classification: a comparative study. In: Barolli, L. (ed) *Advanced Information Networking and Applications* (pp. 252-265). Springer Nature, Switzerland. https://doi.org/10.1007/978-3-031-57916-5_22.

Aslan, O., & Yilmaz, A.A. (2021). A new malware classification framework based on deep learning algorithms. *IEEE Access, 9*, 87936-87951. https://doi.org/10.1109/access.2021.3089586.

Awan, M.J., Masood, O.A., Mohammed, M.A., Yasin, A., Zain, A.M., Damaševičius, R., & Abdulkareem, K.H. (2021). Image-based malware classification using vgg19 network and spatial convolutional attention. *Electronics, 10*(19), 2444. https://doi.org/10.3390/electronics10192444.

Chauhan, P., Mandoria, H.L., & Negi, A. (2021). A review: security and privacy defensive techniques for cyber security using deep neural networks (DNNs). In: Kaushik, K., Tayal, S., Bhardwaj, A., & Kumar, M. (eds) *Advanced Smart Computing Technologies in Cybersecurity and Forensics*. CRC Press, Boca Raton, Florida, pp. 11-22.

Chen, Z., & Cao, J. (2023). VMCTE: visualization-based malware classification using transfer and ensemble learning. *Computers, Materials and Continua*, *75*(2), 4445-4465. https://doi.org/10.32604/cmc.2023.038639.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S. and Uszkoreit, J., (2020). An image is worth 16x16 words: transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Dwivedi, A., Pant, R.P., Pandey, S., & Kumar, K. (2018). Internet of things' (IoT's) impact on decision oriented applications of big data sentiment analysis. In *2018 3rd International Conference on Internet of Things: Smart Innovation and Usages, IoT-SIU* (pp. 1-10). IEEE. Bhimtal, India. https://doi.org/10.1109/iot-siu.2018.8519922.

Elaraby, N., Barakat, S., & Rezk, A. (2024). A generalized ensemble approach based on transfer learning for Braille character recognition. *Information Processing and Management*, *61*(1), 103545. https://doi.org/10.1016/j.ipm.2023.103545.

El-Sayed, R., El-Ghamry, A., Gaber, T., & Hassanien, A.E. (2021). Zero-day malware classification using deep features with support vector machines. In *2021 IEEE 10th International Conference on Intelligent Computing and Information Systems* (pp. 311-317). IEEE. Cairo, Egypt. https://doi.org/10.1109/icicis52592.2021.9694256.

Fayaz, M., Dang, L.M., & Moon, H. (2024). Enhancing land cover classification via deep ensemble network. *Knowledge-Based Systems*, *305*, 112611. https://doi.org/10.1016/j.knosys.2024.112611.

Freitas, S., Duggal, R., & Chau, D.H. (2022). MalNet: a large-scale image database of malicious software. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management* (pp. 3948-3952). ACM, New York, USA. https://doi.org/10.1145/3511808.3557533.

Go, J.H., Jan, T., Mohanty, M., Patel, O.P., Puthal, D., & Prasad, M. (2020). Visualization approach for malware classification with ResNeXt. In *2020 IEEE Congress on Evolutionary Computation* (pp. 1-7). IEEE. Glasgow, UK. https://doi.org/10.1109/cec48606.2020.9185490.

Habib, F., Shirazi, S.H., Aurangzeb, K., Khan, A., Bhushan, B., & Alhussein, M. (2024). Deep neural networks for enhanced security: detecting metamorphic malware in IoT devices. *IEEE Access*, *12*, 48570-48582. https://doi.org/10.1109/access.2024.3383831.

Husain, S., Gupta, S.C., Chand, M., & Mandoria, H.L. (2010). A proposed model for intrusion detection system for mobile adhoc network. In *International Conference on Computer and Communication Technology, ICCCT-2010* (pp. 99-102). Allahabad, India. https://doi.org/10.1109/iccct.2010.5640420.

Jung, I., Lim, J., & Kim, H.K. (2021). PF-TL: payload feature-based transfer learning for dealing with the lack of training data. *Electronics, 10*(10), 1148. https://doi.org/10.3390/electronics10101148.

Krishna, T.B., & Kokil, P. (2024). Standard fetal ultrasound plane classification based on stacked ensemble of deep learning models. *Expert Systems with Applications*, *238*(E), 122153. https://doi.org/10.1016/j.eswa.2023.122153.

Kumar, K., & Dwivedi, A. (2017). Big data issues and challenges in 21st century. *International Journal on Emerging Technologies* (*Special Issue NCETST-2017*), *8*(1) 72-77.

Lu, Q., Zhang, H., Kinawi, H., & Niu, D. (2022). Self-attentive models for real-time malware classification. *IEEE Access*, *10*, 95970-95985. https://doi.org/10.1109/access.2022.3202952.

Majiid, M.R.N., Fredyan, R., & Kusuma, G.P. (2023). Application of ensemble transformer-RNNs on stock price prediction of bank central Asia. *International Journal of Intelligent Systems and Applications in Engineering*, *11*(2), 471-477.

John, S.T., & Swanston, B. (2024). *Cybersecurity stats: facts and figures you should know*. Available: https://www.forbes.com/advisor/education/it-and-tech/cybersecurity-statistics. Accessed in September 2024.

Mohammed, A., & Kora, R. (2023). A comprehensive review on ensemble deep learning: opportunities and challenges. *Journal of King Saud University - Computer and Information Sciences*, *35*(2), 757-774. https://doi.org/10.1016/j.jksuci.2023.01.014.

Nataraj, L., Karthikeyan, S., Jacob, G., & Manjunath, B.S. (2011). Malware images: visualization and automatic classification. In *Proceedings of the 8th International Symposium on Visualization for Cyber Security* (pp. 1-7). ACM, New York, USA. https://doi.org/10.1145/2016904.2016908.

O'Shaughnessy, S., & Sheridan, S. (2022). Image-based malware classification hybrid framework based on space-filling curves. *Computers and Security*, *116*, 102660. https://doi.org/10.1016/j.cose.2022.102660.

Oliva, A., & Torralba, A. (2001). Modeling the shape of the scene: a holistic representation of the spatial envelope. *International Journal of Computer Vision*, *42*(3), 145-175. https://doi.org/10.1023/a:1011139631724.

Pujari, S., Mandoria, H.L., Shrivastava, R.P., & Singh, R. (2022). To identify malware using machine learning algorithms. In: Chaubey, N., Thampi, S.M., Jhanjhi, N.Z. (eds) *Computing Science, Communication and Security.* Springer International Publishing, Cham, pp. 117-127. https://doi.org/10.1007/978-3-031-10551-7_9.

Puneeth, S., Lal, S., Pratap Singh, M., & Raghavendra, B.S. (2024). RMDNet-deep learning paradigms for effective malware detection and classification. *IEEE Access*, *12*, 82622-82635. https://doi.org/10.1109/access.2024.3403458.

Rai, M., & Mandoria, H.L. (2019). A study on cyber crimes, cyber criminals and major security breaches. *International Research Journal of Engineering and Technology*, *6*(7), 233-240.

Rane, N.L., Paramesha, M., Rane, J., & Kaya, O. (2024). Artificial intelligence, machine learning, and deep learning for enabling smart and sustainable cities and infrastructure. *Artificial Intelligence and Industry in Society*, *5*, 2-25.

Rodríguez, E., Valls, P., Otero, B., Costa, J.J., Verdú, J., Pajuelo, M.A., & Canal, R. (2022). Transfer-learning-based intrusion detection framework in IoT networks. *Sensors*, *22*(15), 5621. https://doi.org/10.3390/s22155621.

Salas, M.P., & de Geus, P.L. (2024). Deep learning applied to imbalanced malware datasets classification. *Journal of Internet Services and Applications*, *15*(1), 342-359. https://doi.org/10.5753/jisa.2024.3907.

Seneviratne, S., Shariffdeen, R., Rasnayaka, S., & Kasthuriarachchi, N. (2022). Self-supervised vision transformers for malware detection. *IEEE Access*, *10*, 103121-103135. https://doi.org/10.1109/access.2022.3206445.

Tan, M., & Le, Q. (2019). Efficientnet: rethinking model scaling for convolutional neural networks. In *Proceedings of the 36th International Conference on Machine Learning* (pp. 6105-6114). PMLR. Long Beach, California, USA. https://proceedings.mlr.press/v97/.

Vasan, D., Hammoudeh, M., & Alazab, M. (2024). Broad learning: a GPU-free image-based malware classification. *Applied Soft Computing*, *154*, 111401. https://doi.org/10.1016/j.asoc.2024.111401.

Vinayakumar, R., Alazab, M., Soman, K.P., Poornachandran, P., & Venkatraman, S. (2019). Robust intelligent malware detection using deep learning. *IEEE Access*, *7*, 46717-46738. https://doi.org/10.1109/access.2019.2906934.

Zhong, F., Chen, Z., Xu, M., Zhang, G., Yu, D., & Cheng, X. (2023). Malware-on-the-brain: illuminating malware byte codes with images for malware classification. *IEEE Transactions on Computers*, *72*(2), 438-451. https://doi.org/10.1109/tc.2022.3160357.