International Journal on Advances in Engineering Technology and Science
Volume: 5, Issue: 1, 2024, e-ISSN: 2455-3131
DOI: 10.5281/zenodo.10711191

Malware Identification Using CNN and Deep Forest with Transfer Learning

Nivedita Wahane[1], Chandan Kumar[2]

[1]MTech Scholar, [2]Assistant Professor
Sanjeev Agrawal Global Educational University, Bhopal

[1]nivedita.pratima@gmail.com, [2]chandan.k@sageuniversity.edu.in

***Abstract*— Malware, also known as malicious software, is a set of code that performs malevolent operations with the sole purpose of harming or taking advantage of the individual, government, device, service, network, or for monetary gains. Thus, it has become a priority to find ways to detect malware as a step towards identifying and preventing the malware attacks.**

Researchers have proposed many different malware detection and classification models using various techniques like static-, dynamic-, visualization-based analysis, Machine Learning, Deep Learning, hybrid (i.e. combining two or more different methods), transfer learning approaches, and more.

In this paper, malware identification methodology is proposed using hybrid deep learning models with transfer learning. After converting the suspected file into grayscale image, the proposed methodology will accomplish the task of Malware Identification (i.e. is the provided input malware or benign?) by using CNN (which will be pre-trained by Transfer Learning) for feature extraction from malicious/ benign file's image and Deep Forest for classification.

This proposed methodology will happen in the hope of, first, achieving better accuracy by training the Deep Learning models for malware identification using Transfer Learning; second, to give the end user with only the needed information of whether the file is infected or not as the information about malware families would be of no use to the regular users; and third, since training the model for both identification and classification task will only increase the pre-training, computational time, and resource consumption, to counter this, this model is proposed.

***Keywords*— Malware detection, Visualization-based detection, transfer learning, CNN, Deep Forest**

I. INTRODUCTİON

Malware, also known as malicious software, is a set of code that performs malevolent operations with the sole purpose of harming the device, service, or network. However, although the existence of malware was shown to be as early as the 1949 [1], the early phase's malware were not developed to harm anyone or steal data compared to now where the exponential growth in the malware industry is for the purpose of launching cyber-attacks to victim machines [2] like deleting and encrypting data, hijacking system resources [3] and more.

Malware evolved over the period of time from basic worms and virus to windows and mail virus, to network worms, to ransomware and rootkits, and now, in today's era, malware can be specially created. Malware is today's era have been labeled as APT (Advanced Persistent Threat) aiming to damage military and war forces [1]. There are various types of malwares that can be found today with different damaging abilities like worms, virus, trojan, adware, rootkit, spyware, ransomware, to name a few.

With the evolution of malware, the various ways of getting infected by it has also evolved. It can spread on any device or network using infected floppy disk, sharing documents over internet or USB, downloading email attachments or anything from untrusted source, visiting and clicking on compromised websites, any clicks on the internet e.g. in social media or any video link, or it can also spread via exploiting device vulnerabilities like Microsoft's Macro language vulnerability in Operating System [1].

There is no sector where Malware cannot show its effects. Malware infects sectors from Healthcare to Education to even Defense sector. With a malicious intent, there's no doubt that the number of financial losses caused by the malware is huge. The rise of COVID-19 has accelerated the spread of Malware as more and more devices were made its victim. Thus, without any doubt, it has become a top priority to find ways to detect and classify malware as a step towards identifying and preventing the malware attacks.

Various methods and tools for malware detection exists like static analysis, dynamic analysis, visualization-, signature-, behaviour-, heuristic-, model checking-, Machine Learning- (K-Nearest Neighbour (KNN), Naïve Bayes, Random Forest, Decision Tree, etc.), Deep Learning- (CNN, DenseNet, ResNet, Deep Forest, etc.), cloud, mobile device, IoT-based, and more.

Researchers are also aiming for hybrid approach – combining static & dynamic, Deep Learning & Machine

Learning, or Machine Learning & Machine Learning, or even Deep Learning & Deep Learning to achieve the goal which has shown an increase in percentage of accurately detecting malware in suspected file.

Unlike static and dynamic techniques, visualization-based analysis supports the faster classification of the malware samples as it does not require an application to be disassembled or executed. [18]. In visualization-based approach, the binary file is converted into a grayscale image and this image is given as input for the purpose of malware identification and classification. Fig. 1 shows the process of converting binary file to image. The height of the image is determined from the size of the file as shown in Fig. 2.

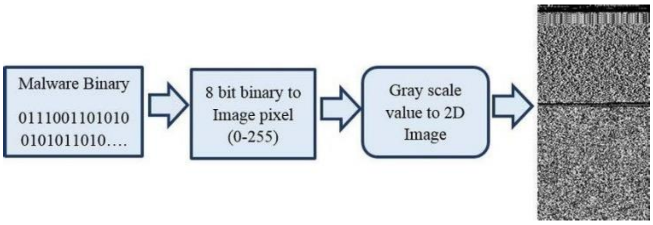
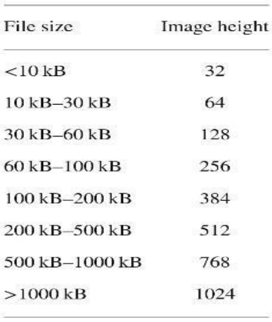


Fig-1. Process of converting a binary file into its corresponding grayscale image



File size	Image height
<10 kB	32
10 kB–30 kB	64
30 kB–60 kB	128
60 kB–100 kB	256
100 kB–200 kB	384
200 kB–500 kB	512
500 kB–1000 kB	768
>1000 kB	1024

Fig-2. Different image height according to different file size

Through malware images, we can find that the images of malware from the same family are visually similar, but images differ in patterns when there is a malware of different family as shown in Fig. 3. Besides, the difference also exists between benign software and malware as shown in Fig. 4. This two knowledge are very useful in terms of malware identification and classification.

The solutions leveraging the combination of visualization-based analysis and Deep Learning have shown the impact lately in the research related to security and privacy. According to [10], the advantage of using Deep Learning over other traditional learning approach is that Deep Learning models can automatically generate high-level features from existing features; it can process very large datasets; it reduces feature space; it increases accuracy, and more to mention a few. Deep Learning has been used in different areas like image processing, computer vision, human action recognition [11], facial emotion recognition, and more. Among other Deep Learning networks, CNNs (Convoluted Neural Networks) are specifically used for image classification [14].

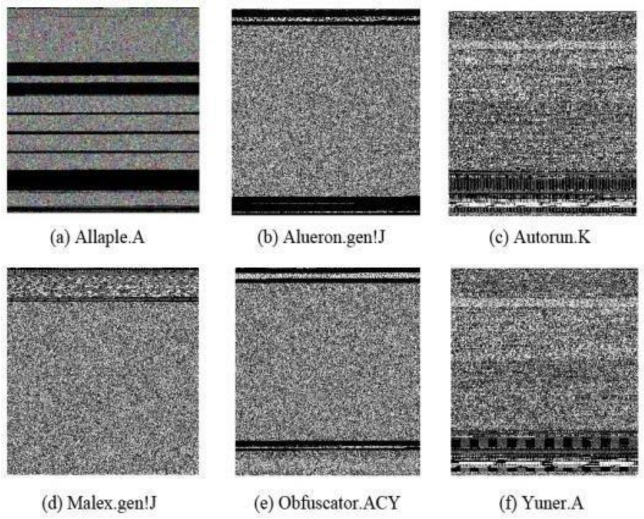

(a) Allaple.A (b) Alueron.gen!J (c) Autorun.K
(d) Malex.gen!J (e) Obfuscator.ACY (f) Yuner.A

Fig-3. Different patterns in grayscale images of different malware families [10]

Fig-4. Greyscale image of a Benign sample [17]

Training a Deep Learning network usually take lot of time and more computational power. For improving model’s performance and reducing time taken for malware identification and classification, a method called Transfer Learning, is also used along with many Deep Learnig models. Transfer Learning helps repurpose a pre-trained model which was trained for say, job A and apply its knowledge to another job B.

Malware can affect any & multiple devices like PCs, IoT, mobile phones, and more. The existing Malware detection and classification approaches could require higher resources in terms of storage and are not suitable for malware detection in resource-constrained devices like IoT. Thus, while some research work already exists, some research in this field for making a lightweight model & for making more efficient malware detection & classification models in general is underway.

II. Related Work

The entire literature review can be viewed as a 4-section review. Section A talks about the basic of Malware and its evolution, section B talks about different approaches and tools that exists in Malware detection along with their performance comparison, section C talks about different Malware detection approaches, and lastly, section D talks about malware classification on resource constrained devices like Android mobile and IoT.

A. Malware Evolution

Cybercriminals develop malware to steal data, bypass access controls, and gain access to a personal computer or harm the target computer, its data, or application. *M. N. Alenezi et. al. [1]* talks about how the malware came into existence as early as 1949 and evolved into special crafted malware. [1] divides the entire evolution timeline into 5 stages – 1949-1991 (Early Phase), 1992-1999 (Second Phase), 2000-2008 (Third Phase), 2005-2016 (Fourth Phase), 2010-Present (Fifth Phase). Each phase talks about various malware created, for what purpose, and how it initially spread, the target and damages caused by it.

B. Tools for Malware Analysis and Their Comparison

R. Samet et. al. [2] talks about a comprehensive review on various Malware detection approaches – Signature-based, Behaviour-based, Heuristic-based, Model checking-based, Deep-Learning based, Cloud-based, Mobile-device based, and IoT based – as well as pros and cons of each approach.

R. Komatwar et. al. [3] categorized malware according to platforms like windows, mobile, etc. [2] also gives us a comparison of static and dynamic analysis approach carried out by various research and also a survey of different research papers that detect malware and classification based on Machine Learning techniques. *O. Aslan [4]* proposed a methodology to learn the well-known malware analysis and detection tools to implement tools on well-known malware and benign programs and compared the obtained results. The test results from [4] indicated that Dynamic malware analysis tool outperformed static analysis tools but it is almost impossible to detect malware by using only one tool and that using static and dynamic analysis tools together increased accuracy and the detection rate. *S. K. Pandey et. al.*

[5] performed a comparison on various malware detection tools – static, dynamic, and online tools. The results obtained verified the results of [4]. *O. Aslan [6]* compared the performance of static malware analysis tools (Peid, PEview, Bintext, MD5deep, Dependency Walker, and IDA Pro) versus antivirus scanners (Norton, McAfee, Kaspersky, Avast, Avira, Bitdefender, and ClamAV) to detect malware. Test results of [6] show that for existing malware, antivirus software detect malware fast and efficient when compared to static analysis tools. However, for unknown malware, static analysis tools performed better than antivirus software.

C. Different Malware Detection Approaches

R. Tian et. al. [7] proposed an approach for distinguishing malicious files from clean files by investigating the behavioural features using logs of various API calls extracted from the executables and apply pattern recognition algorithm and statistical methods (k-fold validation process) to differentiate between files. The result showed that using RF as classifier gives increased accuracy and the proposed method provide an accuracy of 97% in distinguishing malware from cleanware, outperforming comparable previously published techniques. *K. Bhargavi et. al. [8]* introduced a malware identification system based on IoT and IoBT (Internet of Battlefield Things) based on class collection of Op-Code series as a classification feature. For each product a diagram was generated and a deep Eigenspace method for the classification of malware is being used.

A. Ebada et. al. [9] surveyed different visualization-based approaches for malware detection – using Machine Learning, using Deep Learning, and using Hybrid approaches - and compared their accuracy. *A. Yilmaz et. al. [10]* proposed a Deep Learning-based architecture which can classify malware variants based on a hybrid model which integrates two pre- trained network models (AlexNet and ResNet). Their proposed model achieves 97.78% and 94.88% accuracy with MalImg and Big2015 datasets respectively – higher than previous research works. *M. S. Guzel et. al. [11]* proposed a hybrid Deep Learning-based architecture, integrating four pre-trained network models (AlexNet, VGGNet, Google Net, and ResNet) for the recognition and prediction of human actions. According to the experiment conducted by them, this proposed hybrid architecture is more successful than individually executed algorithms.

S. Hussain et. al. [12] used CNN in malware classification and detection through Transfer Learning. EfficientNet model of CNN was used with a compound scaling methodology (scaling width, depth, and resolution of CNN). *S. Kumar et. Al [13]* proposed Malware classification with CNN using Traditional and Transfer Learning. They have trained ResNet with MalImg ang Big2015 datasets from scratch and replaced its last layer with the fully connected dense layer and then out of that layer passed as input for SoftMax layer for classification. This model receives the knowledge from the pre-trained weights of ImageNet model. The pre-trained weights have the knowledge of low-level image features (edges, shapes, corner, and pixel density). This model detected the unknown malware samples without feature engineering which has the extra computational overhead and time-consuming. The model achieved 99.05% accuracy in traditional learning approach and obtained 99.18% accuracy in Transfer Learning approach.

J. Hemalatha et. al. [14] proposed a DenseNet-based Deep Learning model for malware detection. Using DenseNet, they achieved significant performance improvements in classifying malware by handling imbalanced data issues.

The experiments show that the proposed approach can detect new malware samples with higher accuracy (98.23% for MalImg data set, 98.46% for Big2015, 98.21% for MaleVis dataset, and 89.48% for unseen Malicia dataset) and reduced False- Positive rates when compared with conventional malware mitigation techniques while maintaining low computational time. The proposed malware detection solution is also reliable and effective against obfuscation attacks. *S. Roseline et. al. [15]* used Deep Forest technique using sliding window and cascade layering to effectively detect and classify malware. This proposed model achieved a detection rate of 98.65%, 97.2%, and 97.43% for MalImg, Big2015, and MaleVis malware datasets, respectively. The results demonstrate that their proposed solution is effective in identifying new and advanced malware due to its diverse features.

D. Malware Classification on Resource-Constraint Devices

The existing Malware detection and classification approaches could require higher resources in terms of storage and are not suitable for malware detection in resource-constrained devices like IoT. To address this issue,

A. Nag et. al. [16] proposed a custom lightweight CNN for malware image detection. This model achieved 96.64% accuracy. To overcome the challenge of using dynamic analysis in IoT applications, *Z. Zhao et. al. [17]* combined a module of CNN called DEAM (Depthwise Efficient Attention Module) - which can strengthen the attention to the characteristics of malware and improve model effect - with DenseNet to propose a new malware detection and family classification model. This model can reliably detect IoT malware and classify its families as it achieved 98.5% and 97.3% accuracy in terms of family classification for MalImg and Big2015 datasets respectively.

For Android based devices, *J. Singh et. al. [18]* proposed SARVOTAM (Summing of neural architecture and VisualizatiOn Technology for Android Malware identification) where CNN is used to automatically extract rich features from visualized malware thus elimination the feature engineering and domain expert cost. The experiments were done using the DREBIN dataset and the SoftMax layer was substituted with various Machine Learning algorithms like KNN, Random Forest, and SVM (Support Vector Machine). It is observed that CNN-SVM model outperformed original CNN as well as CNN-KNN and CNN- RF. The classification results showed that our method is able to achieve an accuracy of 92.59% using Android certificates and manifest malware images.

X. Zhang et. al. [19] worked towards solving the issue of detecting Android malware in its spread or download stage resulting in early detection of malware before it reaches user's side. They used the network traffic to achieve file-level detection and converted traffic payload into gray-scale images and then in order to obtain more effective features for binary classification and multi-classification task, different methods are applied in two stages. In stage 1, CNN is used for feature extraction and Cascading Deep Forest is used for the binary classification purpose. In stage 2, PCA (Principle Component Analysis) is used for feature extraction and Cascading Deep Forest is used for the multi-classification purpose. Their experimental results show that their model has strong generalization performance, which can not only detect known malware but also unknown malware with high accuracy. *Xu et. al. [20]* proposed a detection framework named Falcon. This method converted the network traffic generated during the running of the application program into 2D images, and extracted features from each image with a pre-trained CNN network. Experiments on the CICAndMal2017 dataset showed that the binary classification accuracy of this method at file-level is 97.16% and that of multi-classification is 84.70%. *Lotfollahi et al. [21]* proposed a Deep Learning-based encrypted traffic detection method called Deep Packet. This method used SAE (Stacked Automatic Encoder) and CNN based model to classify the proposed network traffic and also tell about and group which application produced that packet (Skype, Hangout, etc.) and the particular activity the application was engaged in during the capture session (e.g. voice call, chat, file transfer, or video call)

However, when a file reaches a user's system (PC, mobile, etc.) via any method e.g. USB transfer, email attachment, etc., it is almost impossible to just know if this file contains a malware or not. This is where many different Antiviruses or Malware detection tools and systems come into picture [2 - 21]. Although researchers have proved that the Deep Learning solutions works better than traditional Machine Learning solutions [14], and even hybrid Deep Learning and Deep Learning or Deep Learning and Machine Learning also works better but all these proposed tools and/or systems [11- 21] do both - determine if the file is malicious or not AND classify the malware in its family. The problem with this approach of doing both together could be –

Using both malware and benign samples to train Deep Learning models in order to identify malware and benign input files and also training the same Deep Learning model to classify the found malware with malware samples. This could increase the training time and burden the Deep Learning model. Moreover, according to [19] in the malware categories classification task, the distinguish of classification features become smaller and the high dimensional redundant features are not conducive to improve the classification effect. This again proves that training the same Deep Learning model for feature extraction for both identification and classification task would result in less impact.

From the end user's perspective, the user only needs to know whether the file is infected or not. The information about malware families would be of no use to the regular user along with from where the malicious file came from - an approach proposed in [21].

Although this could be a valuable piece of information in other areas, it is not the case here where any ordinary user is considered. [7] also verified this approach., If the information useful for a device end user is the identification of a file being malicious or not is useful, the proposed methods [11-21] will only increase the pre-training and computational time and resource consumption. As per [19] host-based intrusion detection system (HIDS) residents on the host - the same way that these models are used on device side - it will bring extra consumption on user side, which is difficult to implement on mobile side. And so, for resource constrained devices (like mobile, IoT, etc.), where the device has limited processing and storage capabilities and/ or run on batteries, it becomes necessary that the models used for malware detection and identification keep these constraints in mind.

Compared with the binary classification task of benign and malicious software, in the malware categories classification task, the distinguish of classification features become smaller, and the high dimensional redundant features are not conductive to improving the classification effect.[19]

III. PROPOSED MODEL

Overview

In this section, methodology for identifying malware based on Deep Learnings is proposed. A hybrid approach of combining Deep Learning models is presented here. We will use a hybrid approach, combining different Deep Learning approaches and Transfer Learning method for optimum results. CNN will be used for feature extraction of malicious/ benign file and Deep Forest will be used for classification.

Objectives

The objectives of the proposed methodology are as follows–

To improve accuracy of Malware detection by using Deep Learning models,

To accurately detect the malware in the infected file with less time consumption using Transfer Learning method,

To effectively detect malware for cyber security by hybrid approach of combining different Deep Learning models

The Deep Learning Networks Used

CNN

CNN (Convolutional Neural Network) is a kind of network architecture for deep learning algorithms and is specifically used for image recognition and tasks that involve the processing of pixel data. It is made up of multiple layers, including convolutional layers, pooling layers, and fully connected layers. The convolutional layers are the key component of a CNN, where filters are applied to the input image to extract features such as edges, textures, and shapes. The output of the convolutional layers is then passed through pooling layers, which are used to down-sample the feature maps, reducing the spatial dimensions while retaining the most important information. The output of the pooling layers is then passed through one or more fully connected layers, which are used to make a prediction or classify the image. The architecture of CNN is shown in Fig. 5.

CNNs are trained using a large dataset of labeled images, where the network learns to recognize patterns and features that are associated with specific objects or classes. Once trained, a CNN can be used to classify new images, or extract features for use in other applications such as object detection or image segmentation.

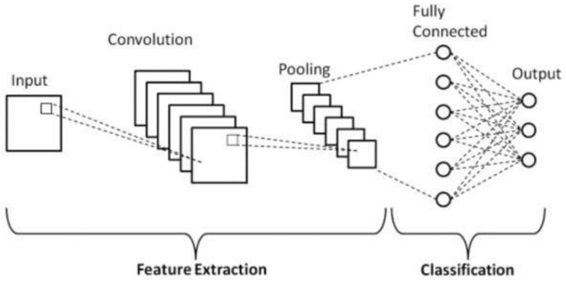


Fig-5. A basic architecture of CNN

Cascading Deep Forest

Deep Forest replaces the neurons of deep neural nets with decision trees. The depth of DF can be dynamically adjusted according to the scale of the data. Thus, it is not only suitable for large-scale data, but also for small-scale and unbalanced data. It includes 2 parts - multi-grained scanning module and cascading module. The multi-grained scanning module uses sliding window with different grains to scan the original data to get input vectors. Number of smaller image instances are generated for further processing. These vectors can be the feature vectors. The final feature vectors are obtained by concatenating class vectors from the sliding window. But since the feature vectors of the malware image has already been generated by CNN, only use the cascade layer for classification purpose. In the cascade layering stage, the resulting feature vector from the first layer is given as input to the four ensemble forests in the second layer. The class vectors generated from each ensemble are concatenated with the feature vector obtained from the first sliding window. This process repeats in an alternative manner. Odd layers take feature vector obtained from the first sliding window, and even layers take feature vector obtained from the second sliding window. The feature vectors from the two sliding windows are alternatively concatenated with the class vectors generated from the four ensembles of each layer. At each layer, validation is carried out to check the accuracy achieved and decide whether this process should proceed further or not. If there is an increase in accuracy, the next layer is processed. If there is no improvement in accuracy, the layering stops. At the optimum layer, the class probability vectors from the four forests of the final layer are averaged to obtain the final prediction probability vector. The class which shows the highest probability is the correct matching class for the given input. The architecture of CaDF (Cascading Deep Forest) is shown in Fig. 6.

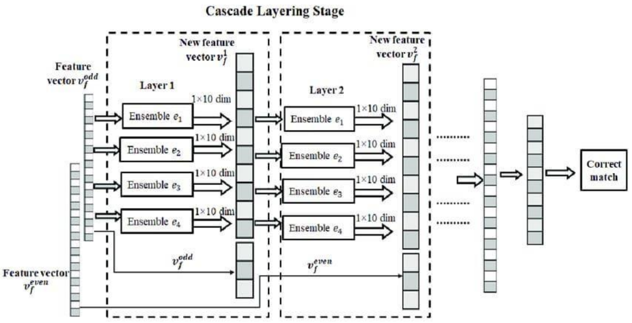


Fig-6. A basic architecture of Cascading Layer of Deep Forest [15]

Proposed Working of the Model

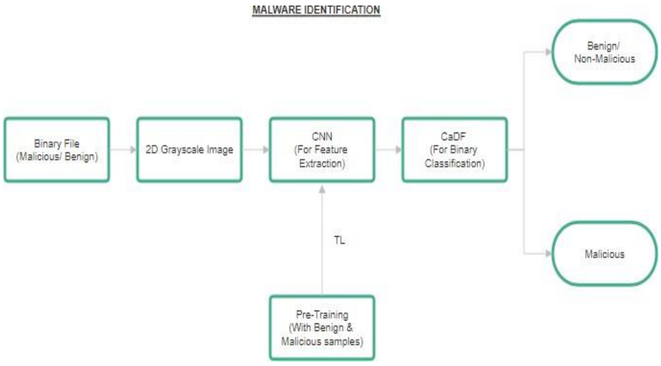

MALWARE IDENTIFICATION
Binary File (Malicious/ Benign)
2D Grayscale Image
CNN (For Feature Extraction)
CaDF (For Binary Classification)
Benign/ Non-Malicious
Malicious
TL
Pre-Training (With Benign & Malicious samples)

Fig-7. Proposed Malware Identification Model

The proposed model of hybrid approach will execute as per the following steps:

STEP 1:

Firstly, the suspected binary files (malware/ benign) will be converted into grayscale images using visualization-based technique as shown in Fig. 1. The file, consisting of bytes, will be first divided into substrings of 8-bit length i.e. Byte. Then we will convert each byte into its respective decimal value between 0 and 255 using the formula:

$$D = B7*2^7 + \qquad + B2*2^2 + B1*2^1 + B0*2^0$$

where B0, B1,...B7 are the 1st bit, 2nd bit,. 8th bit of the

8-bit substring. After this, the file will be converted into 1D vector of decimals and then into 2D matrix with width as per the file size. Table 1 shows different image width for different image sizes. And then the image will be formed using this matrix where 0 denotes black, 255 denotes white and the numbers in between represent various shades of gray. Grayscale images for different malware families are shown in the Fig-3.

STEP 2:

In this step, the image obtained from step 1 is fed as an input to the CNN. This network will already be pre-trained on datasets samples including both malware and benign so that this knowledge will be used as Transfer Learning for the current input. In this stage, we will use CNN for feature extraction from the input image.

STEP 3:

In the 3rd step, we will use the Cascading Deep Forest approach of Deep Learning for identifying the input as malware or benign. The softmax layer of CNN will be substituted for the Cascading Deep Forest as our classifier.

After successful completion of step 1, 2, and 3, the user will get a prompt stating that the selected file is malicious or clean.

Expected Outcomes

The expected outcomes of this research work will include the following-

Improved accuracy of Malware detection from the data samples (that include both malware and benign) by using Deep Learning models as compared to Machine Learning models,

The proposed model should accurately detect the malware in the infected file with less time consumption using Transfer Learning method,

Effective detection of malware for cyber security by hybrid approach of combining different Deep Learning Models as compared to using Deep Learning model or hybrid approach of using Deep Learning & Machine Learning

This model is hoped to detect zero-day malware just like any other malware from the visualization-based approach without classifying it into malware families.

IV. CONCLUSİONS

Although there have been several research done and still ongoing research in the area of malware detection to effectively fight against malware attacks, there is still a long way to go. This paper proposed a Deep Learning hybrid method to identify malware. A hybrid approach of combining Deep Learning models is presented here. We will use a hybrid approach, combining different Deep Learning approaches and Transfer Learning method for optimum results. CNN will be used for feature extraction of malicious/ benign file and Deep Forest will be used for classification. This proposed model is hoped to achieve improved accuracy of Malware detection from the data samples (that include both malware and benign) by using Deep Learning models as compared to Machine Learning models; secondly the proposed model should accurately detect the malware in the infected file with less time consumption using Transfer Learning method; thirdly, effective detection of malware for cyber security by hybrid approach of combining different Deep Learning Models is hoped as compared to using Deep Learning model or hybrid approach of using Deep Learning & Machine Learning.

REFERENCES

[1] Alenezi, M.N., Alabdulrazzaq, H.K., Alshaher, A.A. and Alkharang, "Evolution of Malware Threats and Techniques: a Review", International Journal of Communication Networks and Information Security (IJCNIS), Vol. 12, No. 3, Apr. 2022

[2] Ö. A. Aslan and R. Samet, "A Comprehensive Review on Malware Detection Approaches,", IEEE Access, vol. 8, pp. 6249- 6271, 2020.

[3] Rupali Komatwar & Manesh Kokare, "A Survey on Malware Detection and Classification", Journal of Applied Security Research, Vol. 16, No. 3, pp. 390-420, 2021

[4] Ö. Aslan and R. Samet, "Investigation of Possibilities to Detect Malware Using Existing Tools", IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA), Hammamet, Tunisia, pp. 1277-1284, 2017

[5] S. K. Pandey and B. M. Mehtre, "Performance of malware detection tools: A comparison," IEEE International Conference on Advanced Communications, Control and Computing Technologies, Ramanathapuram, India, pp. 1811-1817, 2014

[6] Ö. Aslan, "Performance comparison of static malware analysis tools versus antivirus scanners to detect malware", International Multidisciplinary Studies Congress (IMSC), 2017.

[7] R. Tian, R. Islam, L. Batten and S. Versteeg, "Differentiating malware from cleanware using behavioural analysis", 5th International Conference on Malicious and Unwanted Software, Nancy, France, pp. 23-30, 2015.

[8] K. Bhargavi, N. Vadivelan, Sarangam Kodati, M. Nalini; "Secure internet of battlefield from malicious software using deep eigenspace learning" AIP Conf. Proc. 2405 (1): 030002, 2022

[9] A. Moawad, A. I. Ebada and A. M. Al-Zoghby, "A survey on visualization-based malware detection," Journal of Cyber Security, vol. 4, no.3, pp. 169–184, 2022.

[10] Ö. Aslan and A. A. Yilmaz, "A New Malware Classification Framework Based on Deep Learning Algorithms," IEEE Access, vol. 9, pp. 87936-87951, 2021

[11] A. A. Yilmaz, M. S. Guzel, E. Bostanci and I. Askerzade, "A Novel Action Recognition Framework Based on Deep-Learning and Genetic Algorithms," IEEE Access, vol. 8, pp. 100631- 100644, 2020

[12] Musaad Darwish AlGarni, Roobaea AlRoobaea, Jasem Almotiri, Syed Sajid Ullah, Saddam Hussain, 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

[13] Sudhar and Sushil Kumar, "MCFT-CNN: Malware Classification with Fine-tune Convolution Neural Networks using Traditional and Transfer Learning", Future Generation Computer Systems, volume 125, issue Dec, pp 334–351, 2021

[14] Hemalatha, Jeyaprakash, S. Abijah Roseline, Subbiah Geetha, Seifedine Kadry, and Robertas Damaševičius, "An Efficient DenseNet-Based Deep Learning Model for Malware Detection", Entropy 23, no. 3: 344, 2021

[15] S. A. Roseline, S. Geetha, S. Kadry and Y. Nam, "Intelligent Vision-Based Malware Detection and Classification Using Deep Random Forest Paradigm", IEEE Access, vol. 8, pp. 206303- 206324, 2020

[16] Ashlesha Hota, Subir Panja, and Amitava Nag, "Lightweight CNN-based malware image classification for resource- constrained applications", Innovations in Systems and Software Engineering, July, pp. 1-14, 2022

[17] Wang Changguang, Zhao Ziqiu, Wang Fangwei, and Li Qingru, "A Novel Malware Detection and Family Classification Scheme for IoT Based on DEAM and DenseNet", Security and Communication Networks; London Vol. 2021, 2021

[18] Singh, Jaiteg, Deepak Thakur, Farman Ali, Tanya Gera, and Kyung Sup Kwak, "Deep Feature Extraction and Classification of Android Malware Images", Sensor Data Fusion Based on Deep Learning for Computer Vision and Medical Applications, no. 24: 7013, 2020

[19] X. Zhang, J. Wang, J. Xu and C. Gu, "Detection of Android Malware Based on Deep Forest and Feature Enhancement", IEEE Access, vol. 11, pp. 29344-29359, 2023

[20] P. Xu, C. Eckert, and A. Zarras, "Falcon: Malware Detection and Categorization with Network Traffic Images", Artificial Neural Networks and Machine Learning – ICANN 2021 September, pp. 117–128, 2021

[21] M. Lotfollahi, M.J. Siavoshani, R.S. Hossein Zade, and M. Saberian, "Deep packet: a novel approach for encrypted traffic classification using deep learning", Soft Computing - A Fusion of Foundations, Methodologies and Applications, Volume 24, Issue 3, Feb, pp. 1999–2012, 2023