DATA SCIENCE TOOLBOX: PYTHON PROGRAMMING PROJECT REPORT

(Project Semester January-April 2025)

MALWARE CLASSIFICATION USING MACHINE LEARNING

Submitted by: Ansh ojha

Registration No: 12312163

Program and Section: P132, K23Sg

Course Code: INT375

Under the Guidance of: Manpreet sir

Discipline of CSE/IT

Lovely School of Computer Science

Lovely Professional University, Phagwara

Malware Classification Using Machine Learning

Abstract— Cybersecurity is increasingly important in an era where technology is prevalent and vulnerable devices are integral to daily life. With the advent of new technologies such as artificial intelligence (AI), evolving cyber threats require innovative and dynamic solutions. One of these solutions is the automatic classification of malware within a system using AI, deep learning (DL), and machine learning (ML). In this paper, it is proposed to improve the reliability of malware detection through a modified multi-agent solution for the automatic classification of malware. The Malimg dataset consisting of twenty-five different classes of malware that have been turned into images is used. The proposed cascaded DL model represents an advancement over previous models on the same dataset, achieving a 97.7% accuracy.

Keywords—Deep learning, Malware classification, multi-agent classifiers

INTRODUCTION:

Malware is a prevalent part of people's lives. Whether people have been a victim before or must go through training at work to keep their devices and systems safe, everyone is familiar. Even with precautions taken, people fall victim to malware. According to the average data leak costs a company about $4.45 million in 2023. With such extreme costs it can be seen why the detection of malware before it can do harm is so important.

Understanding this importance, many researchers have attempted to improve malware detection and classification with the use of various machine learning models. These models have their downfalls though. In this paper, the issue of misclassification due to a lack of training data is investigated with the proposed solution of deep learning ensemble methods.

In researchers explain that the ensemble method in machine learning covers a variety of techniques used to improve the original results of a model, including weighted voting on the original predictions, bagging, boosting, and error correcting the original outputs.

The motivation for developing models for mitigating, preventing, and early detection of cyber threats is apparent and is a top research area. Cybersecurity is important in many domains, including businesses, banks, education, and national security. The continuous evolution of hackers and their utilization of the latest technologies including AI requires the cybersecurity research community, industry, and others to stay one step ahead to protect against cybersecurity and mitigate the consequences even when a cyberattack occurs.

This paper proposes using a second deep learning model to correct errors in an original model's predictions on an undertrained class due to a lack of training data. The public Maling dataset is used. This dataset consists of twenty-five different malware classes. The malware data in this dataset is comprised of malware files converted to grayscale images. This dataset is useful because it represents malware while not actually having malware in the system and can be more accurately analysed with

Convolutional Neural Network (CNN) deep learning models known to perform well on images. One class of this dataset, Autorun's, has very few images and is therefore not predicted with high accuracy. The goal of the researchers is to use the error correcting second model to fix the incorrect classification of Autorun. K after the original model predicts on the entire dataset. Section two of this paper gives a literature review on relevant works. Section three explains the methodology used in the research conducted. Section four explains the results of the research conducted. Lastly, section five draws conclusions on the results of the research and discussed possible routes for future works

OBJECTIVES OF THE PAPER:

To improve the accuracy of malware classification using machine learning, specifically deep learning techniques.

To address the issue of misclassification in underrepresented malware classes (particularly *Autorun.K*).

To develop a multi-agent deep learning approach for automatic malware classification using the Malimg dataset.

To explore ensemble learning techniques by combining models to enhance overall detection reliability.

To leverage image-based malware representation for effective classification using CNN architectures.

BACKGROUND:

Research on the mitigation and prevention of cyber threats is increasing in importance, especially in the era of AI and the Internet of Things (IoT), where it is expected that billions of devices will be connected online in this decade. Researchers are integrating AI in the field of malware detection to explore the use of DL and ML methods for either prevention through early detection or mitigation of the effect of cyber-attacks. In [4] the researchers used three different models on the Microsoft Malware Classification Challenge dataset. Using a Long Short-Term Memory model, they achieved a 97.2% accuracy. Their accuracy increased to 99.4% when using a Convolutional Neural Network model. The researchers then implemented their ensemble Convolutional Neural Network and achieved a 99.8% accuracy. In [5] the researchers used an ensemble Convolutional Neural Network to improve accuracy of both unpacked and packed malware samples. The researchers' proposed method achieved an accuracy of 99% for unpacked malware and 98% accuracy for packed malware. The researchers also reported a low false positive rate and an average of 1.18 seconds for classifying a new sample.

In [6] the researchers used an ensemble machine learning model in the attempt to circumvent current problems in malware classification namely training multiclass models with imbalanced data. The researchers implemented Boost, ExtraTreeClassifier, and combined stacking to construct their ensemble model. This ensemble model achieved 99.72% accuracy.

In [7] the researchers used their novel method to detect malware after converting files to greyscale images and copying the files' opcode sequences. The researchers tested their model on a Microsoft malware dataset and a nine-class malware dataset, achieving 99.88% and 99.36% accuracy respectively.

In [8] the researchers used a multilayered random forest ensemble technique to improve accuracy in malware detection. The researchers' method achieved 98.91% accuracy which was reported as more accurate than the regular deep learning model reference. The researchers also report that their method is more efficient and faster than regular deep learning models.

In [9] the researchers propose the use of convolutional neural networks to analyse malware instead of the common shallow learning algorithms. They converted malware files into grayscale images to be analysed by their convolutional neural network and achieved 98.52% accuracy.

METHODOLOGY:

Based on prior research, the investigation described in this paper was split into three parts: improving on a previously described malware classifying VGG16 model, creating an accurate binary VGG16 model to address an issue with undertraining inherent in the prior model, and the combination of the main VGG16 model with the binary model to improve the model's overall accuracy. VGG16 is a deep convolutional neural network architecture introduced by researchers from the Visual Geometry Group (VGG) at the University of Oxford. It was presented in 2014 as part of the ILSVRC competition and is renowned for its simplicity and depth. VGG16 consists of 16 layers, primarily comprising 3x3 convolutional layers and max-pooling layers. The activation function used is ReLU, enabling the model to learn complex representations effectively. The key characteristic of VGG16 is its uniform architecture, with blocks of convolutional layers stacked on top of each other. This design makes the network easy to understand and modify by increasing the number of layers while maintaining consistent filter sizes. VGG16 achieved impressive results in image recognition tasks and has become a foundational model in the field of computer vision. Its straightforward structure has inspired numerous subsequent deep-learning architectures . Initially, the use of the VGG16 model for classifying malware was explored. This model achieved 97% validation accuracy when tested with the Malimg dataset. A large portion of the errors came from the fact that the classifier always confused two classes of malware. Multiple implementations with variations to training schemes and parameters for VGG16 models were tested, but every time a model was run, it predicted that every Autorun.k malware image was a Yuner.A malware image. A pretrained VGG16 model was also investigated. This pretrained model achieved better accuracy on the Autorun.k and Yuner.A classes, but its overall accuracy was reduced. The original malware classifying VGG16 model was re-tested with different variations to make it more like the pretrained model but none of these attempts yielded an overall accurate model. Subsequently, the creation of a binary classification model that would only predict Autorun.k and Yuner.A was developed. The first binary classification models were made by changing the original VGG16 and pretrained VGG16 models to have two output classes. Further models were then developed in an attempt to find the most accurate binary classification model possible. To aid in the training of these models, the training and validation data was augmented by including rotated versions of images from the referenced dataset to create more data for the models to process. Following the development of the binary classifier, efforts to combine the original malware classifying VGG16 model with the best binary classification model were explored. To create the most accurate overall model, applications of multi agent ensemble deep learning methods were investigated. A depiction of the combined classifier with a primary VGG16 model trained to identify 23 malware classes from the Malimg dataset along with a superclass representing both Yuner.A and Autorun.K, followed by a binary VGG16 classifier to separate the superclass .

RESULTS:

Through prior research, a VGG16 model was tested and performed well, except for the failure to discern Autorun.K from Yuner.A, achieving a 97% test accuracy. Model prediction accuracies for individual malware classes are shown in Table 1. The failure to classify Autorun.K led to attempting to classify the same dataset using a pretrained version of VGG16. The pretrained VGG16 model was able to generate a better accuracy when classifying Autorun.K but not the dataset overall, yielding only a 95% accuracy, as shown in Table 2. This mismatch of results led to the decision to use multiple models together. The goal was to use one VGG16 model to classify only the Autorun.K and Yuner.A variants and to use the original VGG16 model to classify the other twenty-three malware classes that it had already proved most accurate on. The original VGG16 model was already optimized from the prior research so the emphasis was on the development of the VGG16 model to figure out Autorun.K from Yuner.A. First, the original VGG16 model was modified to be a binary classifier solely for the Autorun.K and Yuner.A classes. This model performed like prior implementations and failed to make a distinction between the two classes. The model was then tested using the same activation and optimization functions as the pretrained VGG16 model. This attempt did not yield improved results. The final attempt changed the model’s layer structure to be the same as the pretrained VGG16 model, reducing the number of trainable parameters. Upon the failure of this attempt to make a difference in accuracy, it was decided to investigate using the pretrained VGG16 model to implement the binary classifier. TABLE I. CLASSIFIER ACCURACIES FOR VGG16 MODEL Malware Adialer.C Accuracy Agent.FYI 1 1 Allaple.A Allaple.L The pretrained model immediately yielded better accuracy. After finding the right number of images to train the model on and applying data augmentation techniques, avoiding under and over training, the model was able to predict Yuner.A with 100% accuracy, like in the original VGG16 model, and was also able to improve Autorun.K’s classification accuracy from 0% in the original VGG16 model to 95%. The confusion matrix for this model’s results is shown in Fig. 2. The final, two-stage, multi agent system was tested using ~10% of the images in the dataset, 957 total images. 92 of the images were from the combined Yuner.A and Autorun.K classes and 100% of those were correctly classified into the superclass. Those results were fed through the binary classifier and exactly 1 of the Autorun.K images was misclassified, yielding the accuracy mentioned above. The combined validation accuracy of the overall system is approximately 97.7% with this modification, with a significant increase in the accuracy of classification with respect to the Autorun.K variant.

CONTRIBUTIONS:

- Multi-agent deep learning model using VGG16.
- Addressed class imbalance in malware detection.

- Data augmentation to improve learning.
- Improved malware classification accuracy compared to previous works.
- Practical solution for real-world malware classification systems

CONCLUSIONS:

- Multi-agent deep learning model using VGG16.
- Addressed class imbalance in malware detection.
- Data augmentation to improve learning.
- Improved malware classification accuracy compared to previous works.
- Practical solution for real-world malware classification systems.

FUTURE WORK:

Future work will involve improved automation or pipelining of the models to produce one overall system for classifying malware images. Research plans also include investigations into the creation of additional binary classifier agents or other ensemble methods to improve the overall performance with respect to classification accuracy and time to classify. The likely classes of malware for performance improvement using a multi-agent technique include C2LOP and Swizzor. In examining time to classify for the system, less complex models for the binary classifier agents, models that take advantage of feature reductions and require less complexity than a VGG16 CNN, will be explored.

REFERENCES:

[1] D. Bordered, “Cost of a data breach 2023: Financial industry,” *Security Intelligence*, Aug. 30, 2023. [Online]. Available: https://securityintelligence.com/articles/cost-of-a-data-breach-2023-financial-industry/

[2] T. G. Dietterich, “Ensemble methods in machine learning,” in *Proc. Int. Workshop Multiple Classifier Systems*, Berlin, Germany: Springer, 2000, pp. 1–15.

[3] L. Nataraj, S. Karthikeyan, G. Jacob, and B. S. Manjunath, “Malware images: Visualization and automatic classification,” in *Proc. 8th Int. Symp. Visualization for Cyber Security (Vilseck)*, 2011, pp. 1–7.

[4] B. N. Narayanan and V. S. P. Davuluri, “Ensemble malware classification system using deep neural networks,” *Electronics*, vol. 9, no. 5, p. 721, 2020.