Midterm Project

AI and CyberSecurity DSCI6015

Cloud-based PE Malware Detection API

Yuva Krishna Kishore Inapala

00866841

University of New Haven

Dr. Vahid Behzadan

March 14, 2024

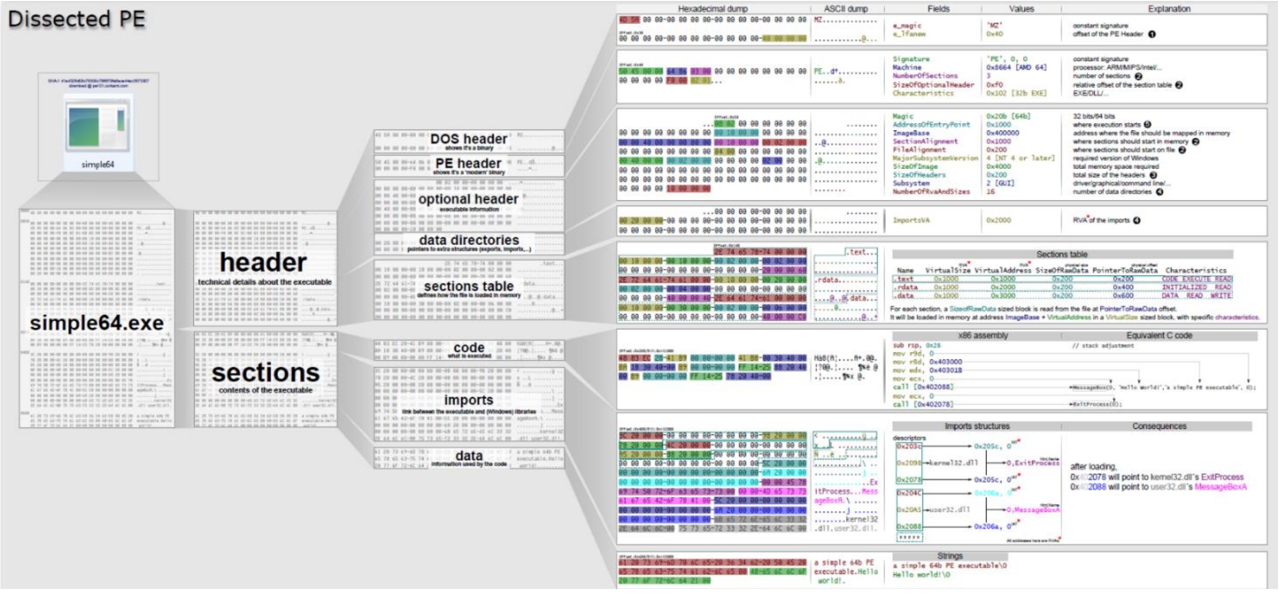
Summary

This report documents the successful development of a cloud-based PE (Portable Executable) malware detection API. The API utilizes a MalConv deep neural network architecture, trained on the EMBER-2018 v2 dataset, to classify Portable Executable (PE) files as malicious or benign.
The project leveraged Google Colab from building and training the model, Amazon EC2 instance for model deployment and web app for creating a user-friendly client application. Python
language was used for the project and the pytorch library was used for the model implementation.

Introduction

PE Files

PE files, utilized by Windows operating systems, serve as a file format for storing executable code and associated data. These files encompass vital details necessary for program execution, encompassing machine instructions, resources, imported libraries, and metadata. Predominantly employed for applications, drivers, and dynamic link libraries (DLLs), PE files adhere to a structured layout featuring headers that furnish insights into the file's attributes, including its architecture, entry point, and section arrangement. Proficiency in comprehending the PE file format proves invaluable for activities such as software analysis, reverse engineering, and malware detection, enabling the examination and modification of executable content.



Malconv

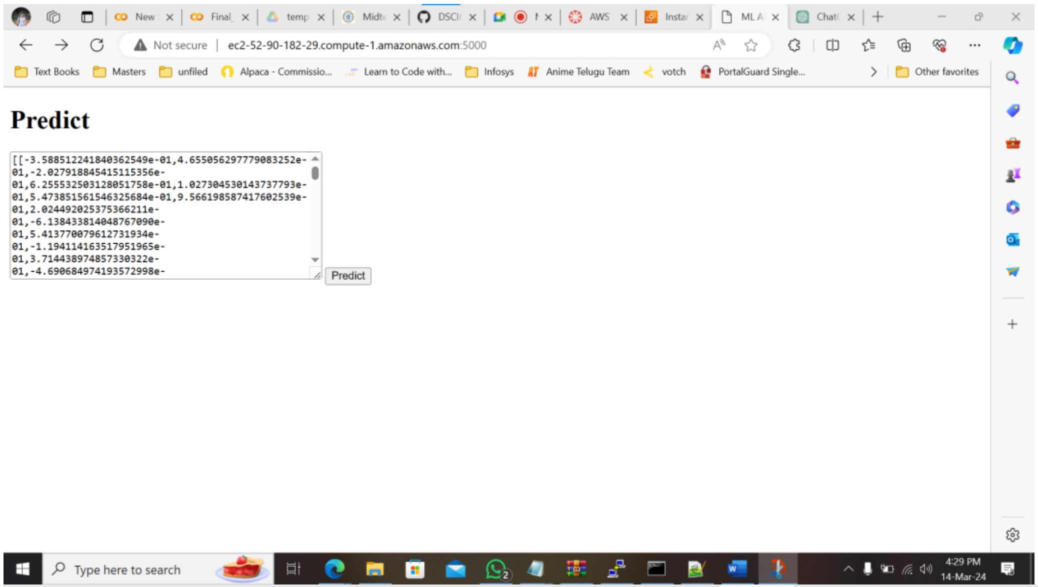
MalConv is a convolutional neural network (CNN) model crafted to identify malicious Windows Portable Executable (PE) files. It utilizes deep learning techniques to scrutinize the raw byte-level content of PE files, extracting significant features and patterns indicative of malicious intent.

The aim of MalConv is to overcome the limitations associated with traditional signature-based malware detection methods. These methods often struggle to keep pace with the constantly evolving landscape of malware threats. By harnessing the capabilities of deep learning, MalConv can learn intricate patterns and correlations within PE files, enabling effective malware detection without relying solely on predefined signatures or heuristics.

This approach offers a more resilient and adaptable solution for detecting previously unseen and sophisticated malware variants.

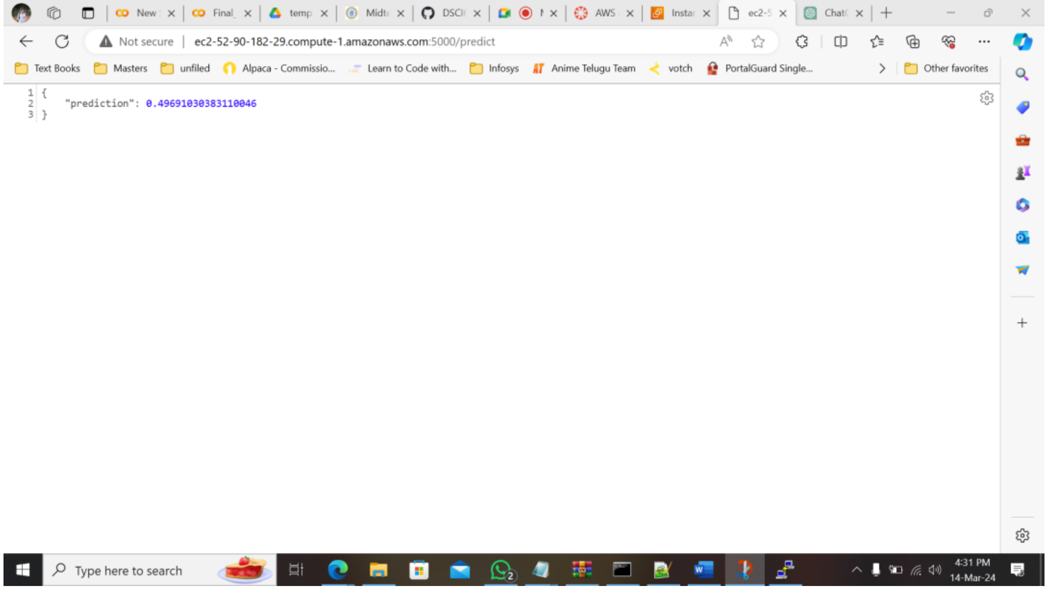
Task Approach:

1. **Building and Training the Model:** A MalConv model was implemented in Python 3.10 using PyTorch 2.x within a Colab Notebook. The model was trained on the EMBER-2018 v2 dataset, achieving significant accuracy in malware classification.
2. **Deploying the Model as a Cloud API:** Amazon EC2 was used to deploy the trained model, creating a cloud-based API for real-time predictions. This process involved leveraging the $100 AWS credit provided through the "AWS Academy Learner Labs" course. Careful cost monitoring ensured adherence to the credit limit. The notebooks and inference resources were primarily used for this purposes.
3. **Creating a Client Application:** A Flask web application was built to provide a user-friendly interface. Users can upload PE files header to the webclient, which are converted by a ember feature extractor in local machine to the deployed API. The application then displays the classification results (malware or benign) where a prediction of probability is displayed if it is less than 0.5 then consider it as Benign else Malware.



Predict

```
[[-3.5885122418403625 49e-01,4.655056297779083252e-
01,-2.027918845415115356e-
01,6.255532503128051758e-01,1.027304530143737793e-
01,5.473851561546325684e-01,9.566198587417602539e-
01,2.024492025375366211e-
01,-6.138433814048767090e-
01,5.413770079612731934e-
01,-1.194114163517951965e-
01,3.714438974857330322e-
01,-4.690684974193572998e-
```


Not secure | ec2-52-90-182-29.compute-1.amazonaws.com:5000/predict

```
{
    "prediction": 0.4969103038311004 6
}
```

Project Results

The project successfully achieved its intended outcomes:

- **Trained MalConv Model:** A well-trained MalConv model capable of classifying PE files as malicious or benign was developed.
- **Deployed Cloud API:** The trained model is deployed on Amazon EC2, functioning as a real-time prediction API accessible via the internet.

Results:

Accuracy: 0.4900

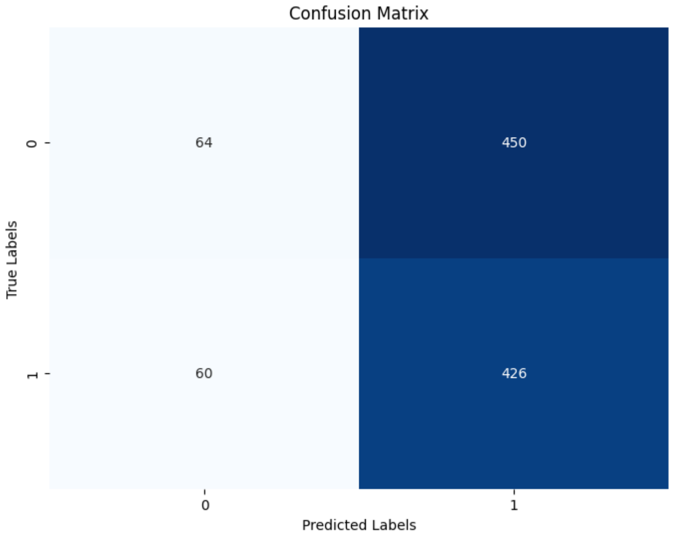
Precision: 0.4863

Recall:0.8765

F1 Score: 0.6255506607929515

Result and Conclusion

Above results suggest that while my model exhibits moderate precision and accuracy, it struggles with recall. A precision of 0.4863 indicates that when the model predicts a positive outcome, it is correct approximately 48.63% of the time. However, the recall of 0.8765 highlights that the model misses a significant portion of actual positive cases. This indicates a potential imbalance in the model's ability to correctly identify relevant instances within the dataset. The F1 score, which combines precision and recall, stands at 0.6255, suggesting a moderate balance between the two metrics. Overall, while my model shows promise, there is room for improvement, particularly in enhancing recall to capture more true positive instances



Conclusion

This objective of the project to successfully develop and deploy a cloud-based PE malware detection API was achieved. The project demonstrates the effectiveness of machine learning for malware classification and the power of cloud platforms like Amazon EC2 and Google Colab for building scalable and user-friendly applications.

Resources:

- https://github.com/endgameinc/ember
- https://github.com/endgameinc/ember/tree/master/malconv
- https://youtu.be/TzW_R36iv48
- https://sagemaker-examples.readthedocs.io/en/latest/intro.html
- https://sagemaker-examples.readthedocs.io/en/latest/frameworks/pytorch/get_started_mnist_train_outputs.html
- https://docs.aws.amazon.com/sagemaker/latest/dg/deploy-model.html
- https://arxiv.org/pdf/1804.04637v2.pdf
- https://youtu.be/ueI9Vn747x4?si=KSFTvR9hBnU0u0DO
- https://youtu.be/oOqqwYI60FI?si=3WKd-iDz93mm1Vbe
- https://youtu.be/g6kQl_EFn84?si=9MHbO9I52AS2pPjx