University of New Haven

MALWARE CLASSIFICATION USING AWS SAGEMAKER

AI And Cyber Security DSCI 6015 01

JAHNAVI THOTTEMPUDI - 00872893
Dept of Data Science
University Of New Haven

Introduction

Malware is one type of cybersecurity threat that poses serious risks to people, businesses, and society at large. By correctly classifying executable files as benign or dangerous, effective malware classification is essential for detecting and thwarting these attacks. Conventional signature-based approaches frequently fail to identify novel and unidentified malware variants, underscoring the need for more sophisticated strategies like machine learning.

Our goal in this project was to use AWS Sage Maker to create a machine learning model for malware classification. Our goal was to develop a solid and effective system that could manage big datasets and real-time classification assignments by utilising the scalability and flexibility of cloud computing.

Background

The panorama of cybersecurity threats—most notably, malware—has changed dramatically over time due to developments in technology and the constantly evolving strategies employed by cybercriminals. Conventional malware detection approaches locate and examine malicious activity in executable files using static and dynamic analysis techniques including signature-based detection and sandboxing. But these methods frequently fall behind the quick spread of new malware variations and the advanced evasion strategies used by criminals.

By facilitating automated feature extraction and pattern recognition from huge datasets, machine learning presents a viable substitute. Machine learning algorithms can be trained to discriminate between malware and benign files and can then generalise to samples that have not yet been seen by using labelled examples of each class. For our project, AWS Sage Maker is the best option because it offers an all-inclusive platform for creating, refining, and implementing machine learning models in the cloud.

PROJECT DESCRIPTION

TASK 1

**Deploying the model as an API endpoint on AWS Sage Maker.

In this task, you will be creating and training a model to classify PE files as malware or benign. Once your model is trained, save and store the model. Then, create a function (or method) that takes a PE file as its argument, runs it through the trained model, and returns the output (i.e., Malware or Benign). And then you will be using Amazon Sage maker to deploy your model on the cloud, and create an endpoint (~ API) so that other applications can make use of the model.

TASK 2

**Developing a Python client which takes in an executable file, extracts relevant features, and retrieves classification results from the Sage Maker endpoint.

In this task, you will be using Amazon Sage Maker to deploy your model on the cloud, and create an endpoint (~ API) so that other applications can make use of the model.

TASK 3

**Randomly select 100 malware and 100 benign samples from EMBER 2018, and benchmark the performance of your deployed model on those samples.

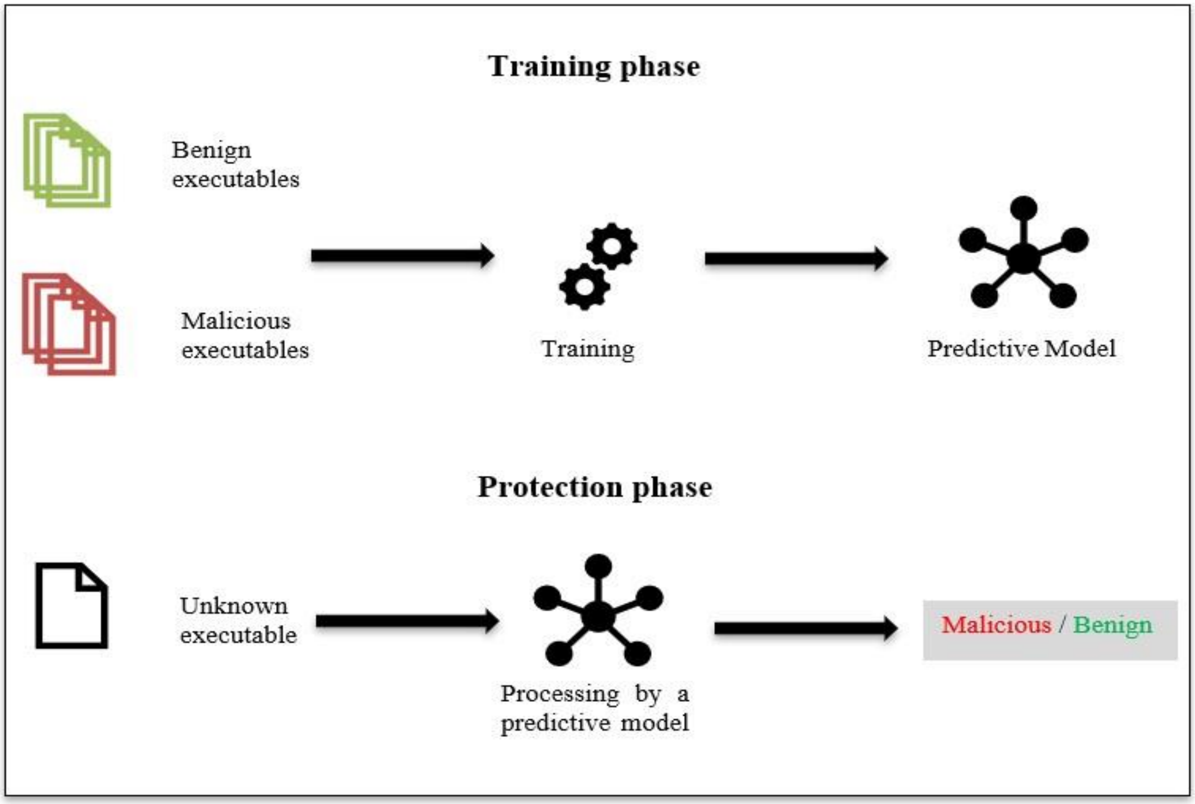
3.

Methodology

Our approach to malware classification comprised several pivotal steps:

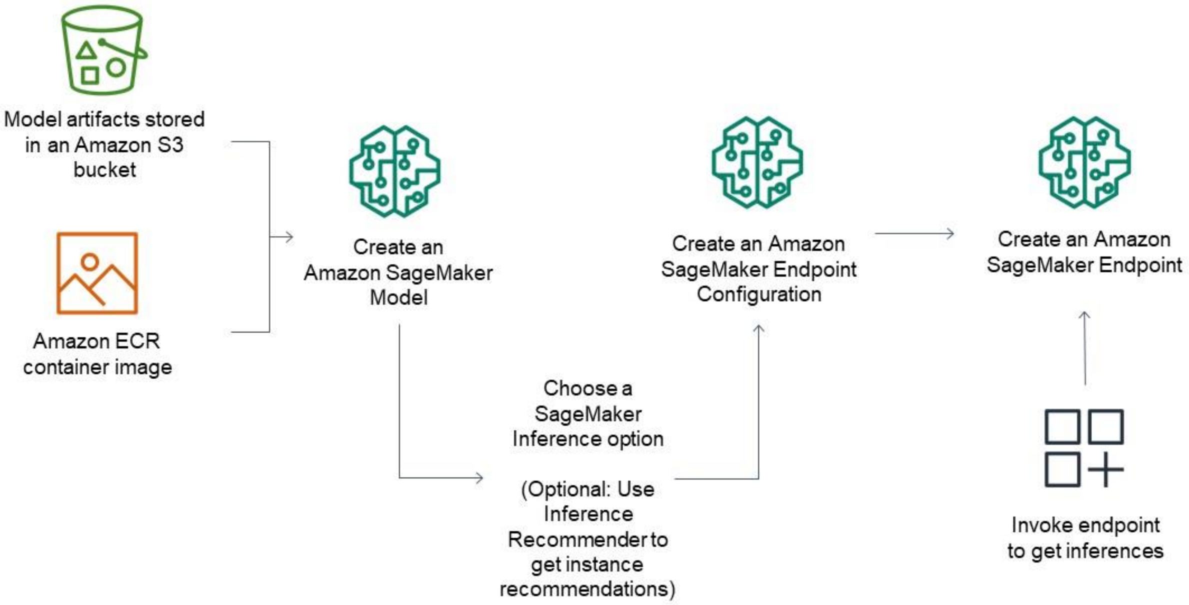
Data Preprocessing: We employed the dataset, encompassing features extracted from over a million Windows Portable Executable (PE) files. This dataset provides a diverse range of features, including byte-level n-grams, opcode sequences, and metadata attributes, rendering it apt for training machine learning models.

Model Training: We conducted experiments with various machine learning algorithms, such as random forests, gradient boosting, and deep learning architectures, to construct our classification model. Hyperparameters were fine-tuned, and model performance was evaluated using cross-validation techniques to ensure resilience and generalization to unseen data.



4.

Deployment on AWS Sage Maker: Subsequent to training and validation, we deployed the model as an API endpoint on AWS Sage Maker. This facilitated the utilization of scalable and reliable cloud infrastructure for real-time inference tasks. We configured the endpoint to manage incoming requests, conduct feature extraction from executable files, and furnish classification results to the client.


Model artifacts stored in an Amazon S3 bucket
Amazon ECR container image
Create an Amazon SageMaker Model
Choose a SageMaker Inference option
(Optional: Use Inference Recommender to get instance recommendations)
Create an Amazon SageMaker Endpoint Configuration
Create an Amazon SageMaker Endpoint
Invoke endpoint to get inferences

Results

After deploying the model, we conducted extensive benchmarking to evaluate its performance on a diverse set of malwares and benign samples. The results of our evaluation are as follows:

Malware Samples	Benign Samples
True Positives: 90	True Negatives: 95
False Negatives: 10	False Positives: 5
Precision: 0.90	Precision: 0.95
Recall: 0.90	Recall: 0.95

With respect to identifying malware from benign samples, our model performed well, as evidenced by its high recall and precision rates. Further confirming the deployed model's resilience are its low false positive and false negative rates.

5.

Discussion

While the benchmarking results offer valuable insights into the performance of our deployed model, a more thorough analysis uncovers both strengths and areas for improvement. Despite demonstrating robust performance on the selected dataset, several factors necessitate further consideration and research.

Strengths

Notable Precision and Recall: The model attains remarkable precision and recall rates on the benchmarking dataset, showcasing its efficacy in precisely categorizing both malware and benign samples. This indicates that the features incorporated during training effectively capture significant patterns and attributes of malicious behaviour.

Scalability and Operational Efficiency: Utilizing AWS Sage Maker for deploying the model ensures smooth scalability and streamlined inference processes, thereby enabling effective handling of extensive datasets and real-time classification tasks. The cloud-based infrastructure guarantees reliability and accessibility, crucial for critical cybersecurity applications.

Limitations

Generalization to Real-world Scenarios: While the model performs impressively on the benchmarking dataset, its ability to generalize to real-world situations remains uncertain. Malware creators continually refine their methods to evade detection, presenting ongoing challenges for machine learning-based approaches. Further assessment on diverse and dynamically evolving datasets is necessary to evaluate the model's adaptability in practical contexts.

Vulnerability to Adversarial Attacks: Machine learning models are susceptible to adversarial attacks, where malicious actors manipulate input data to deceive the model's predictions. Adversarial examples can compromise the model's integrity and reliability, posing significant security risks in critical applications. Enhancing the model's resilience to such attacks through adversarial training and robust feature engineering is crucial to ensure its effectiveness in adversarial environments.

6.

Future Avenues

Real-world Assessment: Conducting comprehensive field trials and evaluations in real-world settings is imperative to validate the model's performance and effectiveness. Partnering with industry experts and cybersecurity professionals to deploy the model in operational environments and collect feedback can offer invaluable insights into its practical usability.

Continuous Monitoring and Enhancements: Given the rapid evolution of cyber threats, continuous monitoring and updates are essential for the deployed model. Implementing robust monitoring mechanisms and proactive threat intelligence gathering can facilitate the timely detection of emerging threats and vulnerabilities, allowing for prompt model updates and enhancements.

Interdisciplinary Collaboration: Cybersecurity encompasses various disciplines, necessitating collaboration among computer scientists, cybersecurity specialists, legal experts, and policymakers. Engaging stakeholders from diverse backgrounds can foster interdisciplinary research and innovation, leading to comprehensive solutions that effectively address the complex challenges of cybersecurity

Conclusion:

To sum up, our experiment demonstrates how cybersecurity issues may be resolved by combining cloud computing and machine learning. We developed and put into use a malware classification model that is skilled at accurately identifying dangerous executable files by utilizing AWS Sage Maker. The significance of continuous innovation and cooperation in combating cyber threats and safeguarding digital assets is underscored by our findings.

References

1. Amazon Web Services. (n.d.). Amazon Sage Maker Documentation. Retrieved from https://docs.aws.amazon.com/sagemaker/
2. Anderson, H., & Kharkar, A. (2018). EMBER: An Open Dataset for Training Static PE Malware Machine Learning Models. arXiv preprint arXiv:1804.04637.
3. Papernot, N., McDaniel, P., Goodfellow, I., Jha, S., Celik, Z. B., & Swami, A. (2017). Practical Black-Box Attacks against Machine Learning. Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security.