Learning Environment Containerization of Machine Learning for Cybersecurity

Zhoulin Li, Hao Zhang
Department of Computer Science
Kennesaw State University, USA
{zli29, hzhang13}@students.kennesaw.edu

Hossain Shahriar
Department of Information Technology
Kennesaw State University, USA
hshahria@kennesaw.edu

Mohammad Masum
Analytics and Data Science Institute
Kennesaw State University, USA
mmasum@students.kennesaw.edu

Dan Lo, Xiaohua Xu
Department of Computer Science
Kennesaw State University, USA
{dlo2,xxu6}@kennesaw.edu

ABSTRACT

Machine learning plays a critical role in detecting and preventing in the field of cybersecurity. However, many students have difficulties on configuring the appropriate coding environment and retrieving datasets on their own computers, which, to some extent, wastes valuable time for learning core contents of machine learning and cybersecurity. In this paper, we propose an approach with learning environment containerization of machine learning algorithm and dataset. This will help students focus more on learning contents and have valuable hand-on experience through Docker container and get rid of the trouble of configuration coding environment and retrieve dataset. This paper provides an overview of case- based hands-on lab with logistic regression algorithm for credit card fraud prediction.

CCS CONCEPTS

•**Security and privacy → Machine Learning, Cybersecurity**.

KEYWORDS

Containerization, Cybersecurity, Machine Learning, Credit Fraud, Hand-on Learning.

1. INTRODUCTION

Machine Learning could help cybersecurity analyze previous cybersecurity threats and detect current security risks. According to Google, 50-70% of emails processed through their Gmail client are spam. Using ML algorithms, Google is making it possible to block such unwanted content with 99% accuracy [7]. Many colleges offer machine learning and cybersecurity courses in their curriculum. However, the integration of machine learning into cybersecurity courses is not common at present. Cyber security experts need to take full advantage of cybersecurity threats.

ACMSE '20, April 2–4, 2020, Tampa, FL, USA

ACM ISBN 978-1-4503-7105-6/20/03
https://doi.org/10.1145/3374135.3385318

The practical activities of network security education can provide all kinds of learners with opportunities to observe and practice, thus benefiting them [4]. Using machine learning to solve cybersecurity problems need to use some related languages, Python is renowned for its concise, readable code and myriad of machine learning algorithm library and became a popular language of machine learning [5]. However, a number of students have troubles in configuring the appropriate python environment and retrieve for the dataset on their own computers. Because each student has a different configuration problem, the instructor spends a lot of time helping students solve configuration problems instead of introducing knowledge related to cybersecurity and machine learning. So, we propose a novel approach with container that can help students get rid of the trouble of configuration environment and focus on the application of machine learning in cybersecurity.

Docker uses operating system-level virtualization to deliver software in packages called containers. These containers are isolated from each other and bind their own libraries, environment and dataset into one image. Running in a container instead of running in a virtual machine Dockerized applications use resources in the host system in a more efficient way than virtual machines [1]. This paper explores Cybersecurity case study by using Docker container. Docker can package everything needed in application into containers which are portable to any system such as Linux and Windows. In this case, we package the implementation of the machine learning algorithm in python, the library we use, and the cybersecurity dataset into one single Docker image (Figure 1). The rest of this article is organized as follows. Section 2 introduces labware design and a specific example. Finally, Section 3 concludes the paper.

2. LABWARE DESIGN

Our labs consist of three parts: pre-lab, hand-on lab, and post add-on lab. Our datasets are real-world cybersecurity datasets.

A. Pre-Lab

The beginning of the pre-lab is learning objectives, which can tell students the purpose of learning this module and what we can learn after this case study.

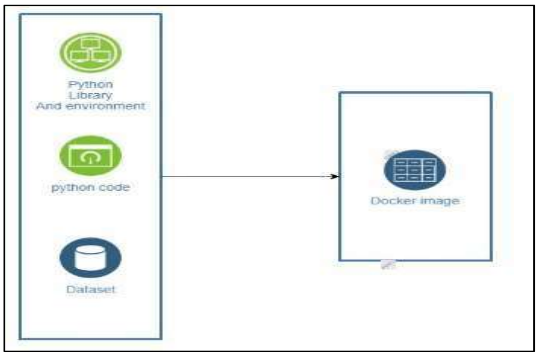

Figure 1: diagram for containerization

Then, students are introduced to a specific cybersecurity case, providing examples of security threats, attack strategies, and attack consequences. The corresponding machine learning algorithms to solve the cybersecurity problems is also introduced. The advantages and disadvantages of the algorithm are also analyzed.

B. Hand-on Lab

In the hand-on lab, the students are provided with a set of step-by-step instructions and detailed explanation of the tasks. In Figure 2, we pull the image from Docker hub. After we input pull request, the image (include machine learning library, cybersecurity dataset, and Python code) will be automatically downloaded to a local computer.

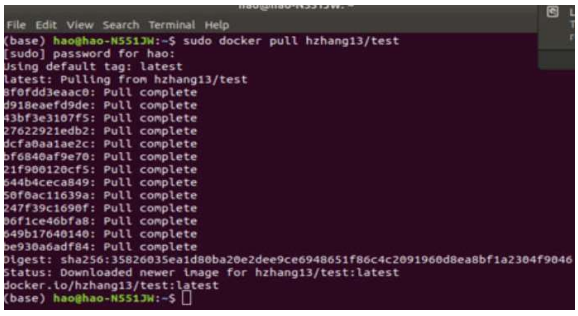

Figure 2: Pulling Docker images

Learners apply logistic regression algorithm to credit card fraud dataset. This dataset comes from Kaggle, and it is an extremely large dataset (284,808 samples) [6]. We split this dataset into training set (75% of the whole dataset) and test set (25% of the whole dataset). After retrieving data, we do feature scaling which can help to normalize data into range (0-1). This will help reduce the computing time of the logistic regression algorithm and improve the accuracy at the same time. Next, we use logistic regression algorithm to train, predict and verify them on test set. Finally, we calculate the accuracy and create a confusion matrix.

This is a screenshot for implementing the Logistic regression algorithm, in order to detect fraudulent credit card transaction in dataset (creditcard.csv). In order to run the lab, students just need to simply type one command line instruction: sudo docker run hzhang13/test and run it on Docker (Figure 3). It shows the analysis results from applying logistic regression algorithm to the test dataset, resulting in very good accuracy and confusion matrix.

Figure 3: Run the lab with Dockerized image

C. Post add-on Lab

In this section, we design challenge questions to promote student creativity and problem solving skills, and to have better understanding of this module. For this case, we left some problems such as re-sampling the minority datasets to improve the performance of fraudulent credit detection; applying Naive Bayes algorithm to detect fraudulent credit and compare the results.

3. CONCLUSION

The overall goal of this work is to provide containerized packaging to enable students to avoid the hassle of downloading libraries and building environments, to address the needs and challenges of ML capacity building in terms of cybersecurity, as well as the lack of teaching materials and real-world practical learning environments. This project can help students quickly understand what should be considered based on the use of ML for cybersecurity issues in a unique way so that students can learn from cases of vulnerabilities. The modules have been used in some related computer science classes and get positive feedback.

REFERENCES

[1] Tozzi, Christopher."Docker: Not Faster than VMs, but More Efficient." Container Journal, Nov. 2016, containerjournal.com/features/docker-not-faster-vms-just- efficient/.

[2] Aune, Nate. "Applications of Docker for Software Training." Appsembler, 2019, www.appsembler.com/blog/docker- applications-software-training/.

[3] McNicol, Lisa. "Announcing the Docker Student Developer Kit & Campus Ambassador Program!" Docker Blog, 12 July2018,www.docker.com/blog/announcing-docker-student- developer-kit-campus-ambassador-program/.

[4] Why Hands-on Skills are Critical in Cyber Security Education, https://www.cybintsolutions.com/hands-on-skills-in-cyber- security-education/, 2018.

[5] Protasiewicz, Jakub. "Why Is Python So Good for AI, Machine Learning and Deep Learning?" Netguru Blog on Python, www.netguru.com/blog/why-is-python-so-good-for-ai- machine-learning-and-deep-learning, 2020.

[6] Andrea Dal Pozzolo, Olivier Caelen, Reid A. Johnson and Gianluca Bontempi. "Calibrating Probability with Undersampling for Unbalanced Classification," Proc. of IEEE Symposium on Computational Intelligence and Data Mining (CIDM), 2015.

[7] Frederic Lardinois, Google Says its Machine Learning tech now blocks 99.9% of Gmail spam and phishing message, https://techcrunch.com/2017/05/31/google-says-its-machine- learning-tech-now-blocks-99-9-of-gmail-spam-and-phishing- message, 2017.