2022 IEEE 46th Annual Computers, Software, and Applications Conference (COMPSAC) | 978-1-6654-8810-5/22/$31.00 ©2022 IEEE | DOI: 10.1109/COMPSAC54236.2022.00087

Scalable Machine Learning Using PySpark

Mohammad Masum
Department of Information Technology
Kennesaw State University
Marietta, USA
mmasum@kennesaw.edu

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

Nazmus Sakib
SUNY Buffalo
nsakib2@buffalo.edu

Maria Valero, Kai Qian, Dan Lo
Kennesaw State University
Marietta, USA
{mvalero, kqian, dlo2}.kennesaw.edu

Fan Wu
Tuskegee University
fwu@tuskegee.edu

Mohammed Karim, Parth Bhavsar
Kennesaw State University
Marietta, USA
{mkarim4, pbhavsar}.kennesaw.edu

Jidong Yang
University of Georgia
jdong.yang@uga.edu

***Abstract*—In this paper, we present a portable labware on Google CoLab for Scalable Machine Learning (SML) with PySpark for facilitating research in Science and Engineering (SML4SE) applications. This will allow researchers to access, share, collaborate, and practice hands-on labs anywhere and anytime without time tedious installation and configuration which will help students more focus on learning concepts and getting more experience in hands-on problem-solving skills for big data analytics.**

Keywords—Scalable Machine Learning, Bigdata Analytics, PySpark, Cybertraining, Portable Labware

I. Introduction

With the ever-increasing size of the data, big data analytics is one of the advanced technologies that promises to deliver better insight from large-scale and highly diverse data. Achieving the highest performance of big data analytics requires selecting appropriate storage and computational frameworks, as well as scalable machine learning techniques [1]. With advanced in-memory programming model and upper-level libraries, Apache Spark has become the de facto framework for big data analytics and eventually adopted as a fast and scalable framework in both academia and industry [1, 2]. In this paper, we present a Scalable Machine Learning (SML) Cyber-training for facilitating research in science and engineering applications.

II. Hands-On Lab: Scalable Machine Learning

The SML cyber infrastructure training system will develop a set of modules aimed at the science and engineering research workforce who use machine learning and big data analytics methods for domain-specific applications or instructional materials on large-scale cyberinfrastructures. Table 1 demonstrates 10 modules associated with the application domains.

Module	Application area of SML
M1	Cybersecurity
M2	Biological Science
M3	Chemical Science
M4	Environmental Science
M5	Science and Mathematics
M6	Mechanical Engineering
M7	Civil Engineering
M8	Electrical Engineering
M9	Industrial Engineering
M10	Software Engineering

Each of the modules will be consist of pre-lab, hand-on lab, and post-lab section. The pre-lab section will include the necessary knowledge for developing the SML models such as problem definition, specification of datasets, and learning outcomes. The hands-on section will demonstrate the step-by-step process for developing the models on the google colab framework, including initiating big data processing tools, data uploading, data preprocessing, feature engineering, model development, model evaluation and result analysis. Finally, the post-lab section will reflect on the real-world problem and share several ideas to extend the research. In addition, we will focus on reproducible and reliable ML in the model development process.

In this paper, we present a scalable machine learning model development step by step process with PySpark to demonstrate a hands-on Lab towards developing the cybertraining system. We use a decision tree classifier, a popular and powerful method in machine learning, to detect malware using ClaMP dataset as an example module for ML with PySpark for cybersecurity.

A. Pre-Lab

The pre-lab gives an overview of the problem definition, data set description and method explanation. Overall, a brief synopsis is provided in the website page (Fig. 1).

978-1-6654-8810-5/22/$31.00 ©2022 IEEE
DOI 10.1109/COMPSAC54236.2022.00087

Figure 1: Pre-Lab of Decision Tree Classifier for Malware Classification

B. Hands-on Lab Practice

The Hand-on section has the step-by-step instruction and some explanation for the coding phase (Fig. 2). Students could practice the hands-on activity lab on their laptop and get the perception of SML solution with PySpark implementation for cybersecurity and gain hands-on experience. In addition, screenshots will be added for each step which help student practice more directional with the visual indication.

Figure 2: Hands-on Lab Practice

The steps in the Hands-on Lab:

Step 0: A user will open Google CoLab Jupyter Notebook (Figure 3)

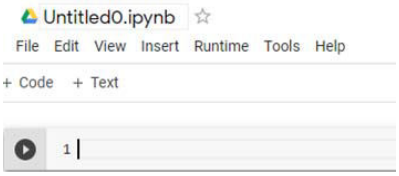

Figure 3: A Google CoLab Jupyter Notebbok

Step 1: Install PySpark and initiate a Spark Session (Fig. 4)

Figure 4: Install PySpark in CoLab

Step 2: Import a dataset (for instance, csv, txt format data) into spark

Step 3: In this step, user will perform explanatory data analysis such as dimension of the data, existence of missing values, number of categorical and numerical features and discover any relationship among features

Step 4: In this step, user will perform data preprocessing including missing value handling, categorical variable conversion, normalizing features, removal of unnecessary features, feature engineering, and split dataset into train & test sets.

Step 5: In this step, user will perform a vector assembler to combine a given list of features into a single vector column which is a requirement for training ML model with PySpark

Step 6: User will train a ML model (e.g., decision tree classifier, random forest classifier, logistic regression, support vector machine, deep neural network, and so on) on the training data and find predictions of the test data

Step 7: User will determine models' performance in terms of different evaluation metrics such as accuracy, f1-score, precision, recall, and auc-roc curve.

Figure 5: Results from PySpark code

III. CONCLUSION

The overall goal of this labware is to address the needs and challenges of developing capacity with Scalable ML for big data analytics in science and engineering applications, as well as to provide a real-world hands-on practice learning environment through effective, engaging case study-based learning approaches.

REFERENCES

1. Salloum, S., Dautov, R., Chen, X., Peng, P. X., & Huang, J. Z. (2016). Big data analytics on Apache Spark. International Journal of Data Science and Analytics, 1(3), 145-164.
2. Shoro, A. G., & Soomro, T. R. (2015). Big data analysis: Apache spark perspective. Global Journal of Computer Science and Technology.