CSC 510 Assignment 3 Report

Justin Oakley

Contents

I. Introduction

This report is about building an understanding of how big data systems function and process big data in a manner suitable for machine learning. The reason why this big data systems are important in the field of data science is due to the continuous evolution of big data systems and resources, so having a fundamental understanding of current systems will assist in the development of this knowledge. There are two main ways of implementing a big data system for big data analysis and machine learning: on a local machine or on a cloud platform, such as Microsoft Azure, Amazon Web Services, or Google Cloud Platform. As for the type of systems utilized in big data science, the Hadoop clusters and Spark clusters are two of the most popular and widely used technologies for data handling and machine learning.

In this assignment, Apache Spark 2.4.0 on an Apache Hadoop cluster will be the type of big data system utilized for machine learning. As for the data that will be processed for machine learning, two network traffic-type datasets, called *nslkdd-version1.csv* and *nslkdd-version2.csv*, were selected. Using these datasets and this big data system, Spark's version of Decision Tree classification was performed for the actual machine learning process to create and refine a model to produce high-quality classification results and to be used for predicting accurate category labels. It should be noted that all coding for the experiment is done in a Jupyter Notebook using the Python 3 language.

II. Machine Learning on the Spark Big Data System

The first step taken in setting up Spark for the data analyzation process of *nslkdd-version1.csv* and *nslkdd-version2.csv* was to install the proper files need for a standalone Spark cluster to be run on an Ubuntu Linux virtual machine and then perform the setup of the environment. Once this is done, the master server and slave servers were launched to

make the Spark cluster ready for data handling. Now that the big data system was fully functional, an iPython Notebook was created, all the appropriate libraries were imported for the application (including a library called ”findspark” which integrates the Jupyter environment with Spark), and a SparkContext was initialized. Following these important steps, the data preprocessing and machine learning was able to be executed.

Since both datasets were known to be of the same source, yet each set had major differences from each other, preprcoessing was mandatory prior to classification, especially because there were only two types of network-traffic that were going to be analyzed: normal-traffic and attack-traffic. With this in mind, the data was read in and any observations that had null values for any of the features, if any had null values, were dropped. It was known that the *nslkdd-version2.csv* columns were a subset of the features of *nslkdd-version1.csv*, so only the intersection of both sets' columns were utilized in the supervised learning. Following this, the label column of *nslkdd-version1.csv* was converted from categorical data to indexed, numerical data; then each observation in both datasets were labeled either as normal-traffic or attack-traffic and the original label columns were dropped since they were no longer needed for classification. In order to obtain accurate data, a set's columns were statistically comparted to the other set's counterpart columns, respectively, and any features with matching stats were kept while features with non-matching stats were dropped because they could not be deemed reliable. Once this was finished, it was recognized that some of the columns that were kept contained a vast amount of zero-values; so it was decided that any columns that had a third-quartile statistic that was zero should be dropped. Finally, the data features were assembled into vectors and standarized. At this point, the data was now primed and ready for Decision Tree classification.

Using Spark's version of a Decision Tree classifier, the data was split into training and testing and both sets were used to fit and transform the modell; because the technique used in the experiment was a Decision Tree, probabilistic measures were used to train the data and qualitative measures are used to test and validate (which will be discussed later) the data.