Big Data (ECE 595-004/007), Fall 2018
Hands-on 4: Spark Programming
Due date: December 07, 2018, 11:59 PM

Tasks

Task 1 (3pts). Spark Installation

a. Download Apache Spark from http://mirrors.advancedhosters.com/apache/spark/spark-2.4.0/spark-2.4.0-bin-hadoop2.7.tgz

b. Extract setup file in the downloaded folder using command: **tar xvfz spark-2.4.0-bin-hadoop2.7.tgz**

c. Move extracted folder to /usr/local/spark using command: **sudo mv spark-2.4.0-bin-hadoop2.7 /usr/local/spark**

d. Change ownership of /usr/local/spark using command: **sudo chown –R bigdata:bigdata /usr/local/spark**

Note: here **bigdata** is user name. In your case, it may be different. You can get the user name by executing command **whoami**

e. Update .bashrc file in your home directory using command: **gedit ~/.bashrc**

Add following lines in **.bashrc** file at the end:

export SPARK_HOME=/usr/local/spark

export PATH=$PATH:$SPARK_HOME/bin/

Close .bashrc file and execute **source .bashrc** command

f. Copy **/usr/local/spark/conf/spark-env.sh.template** file to **/usr/local/spark/conf/spark-env.sh**. Then, add following line in **/usr/local/spark/conf/spark-env.sh**:

export SPARK_DIST_CLASSPATH=$(hadoop classpath)

g. Start Hadoop using **start-all.sh** command and execute **pyspark**.

If you see any output, it means Spark has installed in your system. You may exit from **pyspark** shell.

h. Install **numpy** and **scipy** using following commands.

sudo apt-get install python-pip

sudo pip install numpy scipy

They will be used in Spark ML libraries.

Task 2 (8 pts) Write Spark application that reads "task2.txt" file from HDFS cluster and finds top 10 most frequent words and their frequencies. In the text file, a few words may appear in different forms, e.g. The, the, you have to treat them same. In addition, some words may have double quote, single quote or other non-alphabet character in the prefix or suffix, your program should be able to remove them and then consider the remaining characters as word. Implement this program through RDD **transformation** and **action** operation. You may start with uploaded skeleton code **spark_wc.py** for word count program. To run your spark application, execute **spark-submit <your Spark Python file name>.** Please use **firstname_lastname_task2.py** format for naming the program file.

Task 3 (8 pts) Write Spark application that reads "task3.txt" file from HDFS cluster and finds average and standard deviation of stores' sales in each city. Implement this program through Spark Dataframe and Spark SQL. You may start with the uploaded skeleton code **spark_std.py** for word count program. To run your spark application, execute **spark-submit <your Spark Python file name>.** Please use **firstname_lastname_task3.py** format for naming the program file. You may refer slides and following link:

https://www.analyticsvidhya.com/blog/2016/10/spark-dataframe-and-operations/

Task 4. (**2 pts**) Upload task4.txt file in HDFS cluster and run **spark_ml.reg.py** Spark application. The application is Linear Regression implementation in Spark. The dataset used in task4.txt file is taken from here. You can visit the link to get the description of the dataset. Run the application by varying parameters like maxIter, regParam, and elasticNetParam. Mention your observation in the report. You may refer the following link:
https://www.datacamp.com/community/tutorials/apache-spark-tutorial-machine-learning

Task 5. (2 pts) Upload task5.txt file in HDFS cluster and run **spark_ml_clf.py** Spark application. The application is Logistic Regression implementation in Spark. The dataset used in task5.txt file is taken from here. You can visit the link to get the description of the dataset. Run the application by varying parameters like maxIter and regParam. Mention your observation in the report.

Task 6. (7 pts) Use task5.txt dataset and implement Random Forest model for classification for it. You may modify **spark_ml_clf.py** for the implementation. Compare your results with logistic regression implementation. Please use **firstname_lastname_task6.py** format for naming the program file.
Refer Python implementation in the following link:
https://spark.apache.org/docs/2.2.0/ml-classification-regression.html#random-forest-classifier