Electrical and Computer Engineering, Purdue University Northwest
Big Data (ECE59500/ECE49500)
Hands-on Assignment 3

Task 1 [20 points]. You **must have** to add screenshots in the report for the output of underlined text in orange color to get full score for the task.

Part A [5 points]: Flume installation and set-up.

1. Download Apache Flume from http://www.apache.org/dyn/closer.lua/flume/1.9.0/apache-flume-1.9.0-bin.tar.gz
2. Extract the setup file in the downloaded folder using command:
   `tar xvfz apache-flume-1.9.0-bin.tar.gz`
3. Move the extracted folder to `/usr/local/flume` using command
   `sudo mv apache-flume-1.9.0-bin /usr/local/flume`
4. Change the ownership of `/usr/local/flume` using command
   `sudo chown -R bigdata:bigdata /usr/local/flume`
   Note: here `bigdata` is user name. In your case it may be different. You can get the user name by executing command `whoami`
5. Update `.bashrc` file in your home directory using command: `gedit ~/.bashrc`
   Add following lines in `.bashrc` file at the end:
   ```
   export FLUME_HOME=/usr/local/flume
   export PATH=$PATH:$FLUME_HOME/bin/
   ```
   Close `.bashrc` file and execute `source .bashrc` command
6. Start Hadoop service (if not running) using **`start-all.sh`** command and execute `flume-ng --help`. If you see any output with command options, it means Flume has installed in your system.

Part B [15 points]: Create a Flume agent to copy data files `flume1.txt` and `flume2.txt` (uploaded with the assignment) from `/tmp/flume` directory to `/user/flume` directory in HDFS cluster. You have to copy each file in `/tmp/flume` folder one by one and Flume should copy them in HDFS. Finally verify that files have been copied into the cluster. Show the files content in HDFS cluster. You need to write and save the agent file in `/usr/local/flume/conf` folder.

Task 2 [25 points]. You **must have** to add screenshots in the report for the output of underlined text in orange color to get full score for the task.

Part A [10 points]: Hive installation and set-up

1. Download Hive from https://mirrors.sonic.net/apache/hive/hive-2.3.7/apache-hive-2.3.7-bin.tar.gz
2. Extract the setup file in the downloaded folder using command: `tar xvfz apache-hive-2.3.7-bin.tar.gz`
3. Move the extracted folder to `/usr/local/hive` using command: `sudo mv apache-hive-2.3.7-bin /usr/local/hive`
4. Change the ownership of `/usr/local/hive` using command: `sudo chown -R bigdata:bigdata /usr/local/hive`

5. Update `.bashrc` file in your home directory using command: `gedit ~/.bashrc`
   Add following lines in `.bashrc` file at the end:
   ```
   export HIVE_HOME=/usr/local/hive
   export PATH=$PATH:$HIVE_HOME/bin/
   export HIVE_CONF_DIR=$HIVE_HOME/conf
   ```
   Close `.bashrc` file and execute `source .bashrc` command
6. Copy `/usr/local/hive/conf/hive-env.sh.template` file to `/usr/local/hive/conf/hive-env.sh` and open it using `gedit /usr/local/hive/conf/hive-env.sh` command
   Add `HADOOP_HOME=/usr/local/hadoop` in the file
7. Create `hive-site.xml` inside `/usr/local/hive/conf` folder using command `touch hive-site.xml`
   Open it using `gedit /usr/local/hive/conf/hive-site.xml`
   Add following lines in the file
   ```
   <configuration>
   <property>
           <name>system:java.io.tmpdir</name>
           <value>/tmp/hive/java</value>
   </property>
   <property>
           <name>system:user.name</name>
           <value>${user.name}</value>
   </property>
   <property>
           <name>javax.jdo.option.ConnectionURL</name>

   <value>jdbc:derby:;databaseName=metastore_db;create=true</value>
   </property>
   </configuration>
   ```
8. Copy `hive-default.xml.template` to `hive-default.xml` inside `/usr/local/hive/conf` folder
9. Execute command `schematool -initSchema -dbType derby`
10. Start Hadoop service (if not running) using `start-all.sh` and then execute hive command. If you see `hive` prompt, it means Hive has installed in your system.

Part B [15 points]: Upload `students.txt` (uploaded in with the assignment) file into HDFS cluster. Find the average score for each subject by creating a Hive table for the file and execute query on it.

Task 3 [55 points]. You **must have** to add screenshots in the report for the output of underlined text in orange color to get full score for the task.

Part A [10 points]: Spark installation and set-up

1. Download Apache Spark from https://mirrors.ocf.berkeley.edu/apache/spark/spark-3.0.1/spark-3.0.1-bin-hadoop2.7.tgz
2. Extract setup file in the downloaded folder using command:
   `tar xvfz spark-3.0.1-bin-hadoop2.7.tgz`
3. Move extracted folder to `/usr/local/spark` using command:
   `sudo mv spark-3.0.1-bin-hadoop2.7 /usr/local/spark`
4. 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`
5. 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/
   export PYSPARK_PYTHON=python3
   ```
   Close `.bashrc` file and execute `source .bashrc` command
6. Copy `/usr/local/spark/conf/spark-env.sh.template` file to `/usr/local/spark/conf/spark-env.sh`. Then, add the following line in `/usr/local/spark/conf/spark-env.sh`:
   `export SPARK_DIST_CLASSPATH=$(hadoop classpath)`
7. Start Hadoop (if not running) using `start-all.sh` command and execute `pyspark`. If you see Spark banner in the output, it means Spark has installed in your system. You may exit from `pyspark` shell.

Part B [20 points]: Write Spark application that reads "**words.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_task3b.py` format for naming the program file.

Part C [25 points]: (*Optional for ECE 49500 students*) Write Spark application that reads "sales.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`. To run your spark application, execute `spark-submit <your Spark Python file name>`. Please use `firstname_lastname_task3c.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/