IEMS5730 Spring 2019
Homework 4

Release date: Mar 18, 2019
Due date: 11:59am, Apr 22, 2019
The solution will be posted right after the deadline, so no late homework will be accepted!

Every Student MUST include the following statement, together with his/her signature in the submitted homework.

I declare that the assignment submitted on Elearning system is original except for source material explicitly acknowledged, and that the same or related material has not been previously submitted for another course. I also acknowledge that I am aware of University policy and regulations on honesty in academic work, and of the disciplinary guidelines and procedures applicable to breaches of such policy and regulations, as contained in the website http://www.cuhk.edu.hk/policy/academichonesty/.

Signed (Student_____) Date:_____

Name_____ SID_____

Submission notice:

- Submit your homework via the elearning system

General homework policies:

A student may discuss the problems with others. However, the work a student turns in must be created COMPLETELY by oneself ALONE. A student may not share ANY written work or pictures, nor may one copy answers from any source other than one's own brain.

Each student **MUST LIST** on the homework paper the **name of every person he/she has discussed or worked with**. If the answer includes content from any other source, the student **MUST STATE THE SOURCE**. Failure to do so is cheating and will result in sanctions. Copying answers from someone else is cheating even if one lists their name(s) on the homework.

If there is information you need to solve a problem but the information is not stated in the problem, try to find the data somewhere. If you cannot find it, state what data you need, make a reasonable estimate of its value, and justify any assumptions you make. You will be graded not only on whether your answer is correct, but also on whether you have done an intelligent analysis.

Q1 [20 marks]: Spark Basic RDD

In this question, you are required to install Spark on your cluster set up in Homework#1 and explore some of the basic RDD concepts. You are recommended to reuse the Hortonworks Data Platforms with 4 nodes (VM instances). You can use any programming language supported by Spark but need to operate on the RDD level.

(a) **[10 marks]** Naive implementation of PageRank in Spark. In this part, write your own PageRank program. Then, run your program on the given dataset [3], submit the code and the top 100 nodes.

(b) **[10 marks]** Advanced implementation of PageRank in Spark. In this part, you need to take advantage of pre-partition mechanism to reduce the shuffling overheads. Please adjust the number of partitions (try 3 different cases) and compare their performance. Submit your result and explain your observations.

Q2 [20 marks]: Spark SQL

In this question, we will analyze the report of crime incidents in Washington D.C. . The dataset comes from the District of Columbia's Open Data Catalog. Download the report of crime incidents in 2013 from:

http://opendata.dc.gov/datasets/crime-incidents-in-2013

Upload the data to HDFS. After you explore this csv file, you can find it consists of around 20 columns.

(a) **[5 marks]** We are interested in the following information:

(CCN, REPORTDATETIME, OFFENSE, METHOD, LASTMODIFIEDDATE, DISTRICT).

Use Spark to truncate the file and only keep these 6 items of each line of record.
If these fields are empty in some lines, please filter out those lines.

Hints: if these fields are empty in some lines, please filter out those lines.

(b) **[5 marks]** Use Spark queries[4] to count the number of each type offenses and find which time-slot (shift) did the most crimes occur.

(c) **[10 marks]** The dataset below tracks the crime incidents from 2010 to 2018.

http://opendata.dc.gov/datasets?q=crime%20incidents%20

Merge these 9 tables into one and compute the percentage of gun offense for each year. Discuss the effect of Obama's executive actions on gun control.

Q3 [20 marks]: Spark Streaming

In this question, you are required to identify Twitter trending hashtags using the Sparking Streaming and Twitter API. You are allowed to refer to publicly available Spark examples/ source-codes AS LONG AS you clearly identify the borrowed codes and acknowledge your sources in your submission. You can either use the low-level streaming interface or the Structured Streaming capabilities of Spark.

(a) **[5 marks]** To read tweets, you will need to setup a consumer key+secret pair and an access token+secret pair using your Twitter account. Please follow the instructions [1] to setup these temporary access keys.

(b) **[15 marks]** Refer to [5], [6] and write a program to find the Top-10 most popular hashtags under the topic: "Trump" within the past 10 minutes and update the Top-10 list once every 2 minutes.

Q4 [30 marks]: HBase Setup and Basic operations

(a) **[5 marks]** Please install a fully-distributed HBase on top of Hortonworks Data Platform in Q1. In other words, you cannot just use a single-node cluster.

(b) **[15 marks]** Import the "Google Books 2" dataset in [14] to HBase using Bulk Loading in [15]. It includes two steps: upload the file to HDFS and convert it into HFiles; then use CompleteBulkLoad in [17] to move the generated file into an HBase table. A detailed example is given in [18]. As for the first step, you can either write your own MapReduce code to convert the file to HFiles or use ImportTsv in [16] to convert it. When creating the table, you should make sure the table is splitted into at least 3 parts so that each part is stored on a different node in the cluster.

(c) **[10 marks]** Perform the following tasks (use either Java code or command line shell):

1. Insert the following record into the table
   iems5730 2019 100 11
2. Get all the records in Year 1671 with occurrence larger than 100. (Filter is needed).
3. Delete the records in part 2.

Hints: You may need to add a unique rowkey for each row in the dataset.

Q5 [Bonus 20 marks]: Spark ML

In this question, we will take advantage of the Spark Machine Learning Library MLlib to deal with the Netflix Challenge and implement a Movie Recommender System based on collaborative filtering.

(a) **[10 marks]** The Netflix Challenge is aimed at predicting user ratings for films, based on previous ratings without any other information about the users or films[7]. In this part, you are asked to finish a recommender systems based on collaborative filtering. You can read [8] [9] to find more informations about this method.

Fortunately, Spark MLlib already supports collaborative filtering. You can directly use the ALS (Alternating Least Squares) algorithm [10] in MLlib to fill in the missing entries of a user-item association matrix in GroupLen dataset [6]. Please print out the top 10 favorite movies of user 1, user 1001 and user 10001.

Hints: You do not need to adjust any parameters in the ALS algorithm. You can just use:
val model = ALS.train(ratings, 50, 10, 0.01, 0.01)

(b) **[10 marks]** In this part, you are asked to use cross validation (CrossValidator) in MLlib to evaluate the performance of the model and choose the best set of parameters (in the ALS algorithm) for this recommender system. You can choose any one of MSE(mean square error), ROC(receiver operating characteristic curve) or AUC(Area under the Curve of ROC) as the indicator to compare the performance of part 1 and part 2. You can read [11] [12] for more information.

Hints: Before cross validation, you need to use a ParamGridBuilder to construct a grid of parameters to search for the best set of parameters.

Reference:

[1] Twitter Credential Setup (Section 1.2):
http://ampcamp.berkeley.edu/3/exercises/realtime-processing-with-spark-streaming.html
[2] twitter4j library

http://twitter4j.org/javadoc/twitter4j/TwitterStream.html#filter-java.lang.String...-
[3] SNAP Google Web Data
https://snap.stanford.edu/data/web-Google.html
[4] Spark SQL programming guide:
http://spark.apache.org/docs/1.6.0/sql-programming-guide.html
[5] Spark Streaming Programming Guide:
https://spark.apache.org/docs/1.6.0/streaming-programming-guide.html
[6] Grouplens 20M dataset:
https://grouplens.org/datasets/movielens/20m/
[7] Netflix Prize https://en.wikipedia.org/wiki/Netflix_Prize
[8] coursera collaborative filtering
https://www.coursera.org/learn/machine-learning/lecture/2WoBV/collaborative-filtering
[9] Matrix Factorization methods
http://www.research.att.com/people/Volinsky_Christopher_T/custom_index.html
[10] ALS in MLlib
https://spark.apache.org/docs/latest/mllib-collaborative-filtering.html
[11] Spark Cross Validation
https://spark.apache.org/docs/latest/ml-tuning.html
[12] Introduction of Spark ML
https://zhuanlan.zhihu.com/p/24649048
[13] Google Books 1 :
http://storage.googleapis.com/books/ngrams/books/googlebooks-eng-all-1gram-20120701-a.gz
[14] Google Books 2:
http://storage.googleapis.com/books/ngrams/books/googlebooks-eng-all-1gram-20120701-b.gz
[15] Bulk Loading:
http://hbase.apache.org/0.94/book/arch.bulk.load.html
[16] ImportTsv:
http://hbase.apache.org/0.94/book/ops_mgt.html#importtsv
[17] CompleteBulkLoad:
http://hbase.apache.org/0.94/book/ops_mgt.html#completebulkload
[18] Bulk Loading Example:
http://blogs.msdn.com/b/bigdatasupport/archive/2014/12/12/loading-data-in-hbase-tables-on hdinsight-using-bult-in-importtsv-utility.aspx
[19] Hbase Tutorial:
http://www.tutorialspoint.com/hbase