Big Data Analytics

Assignment # 2 Due Date: 24 April 2022

Assignments are to be done in group of 3. No late assignments will be accepted.

HONOR POLICY

This assignment is a learning opportunity that will be evaluated based on your ability to think, work through a problem in a logical manner. You may however discuss verbally or via email the assignment with your classmates or the course instructor and TA, and use the Internet to do your research, but the written work should be your own. Plagiarised reports or code will get a zero. If in doubt, ask the course instructor.

Task

- You have already loaded the dblp data into the MongoDB in your last assignment. Now you will use that data by connecting mongoDB with Hadoop and Load data in chunks or you can load the data manually using the csv generated from MongoDB of the Data and load it in HDFS File System (this will result in some marks deduction).
- Now you will treat every document of MongoDB as a transaction and Apply SON Algorithm (taught in the class) on that chunk of data. SON Algorithm impart itself well to a parallel – computing environment. Each of the chunk can be treated in parallel, and the frequent Item-sets from each chunk unite to form the candidates. Steps are as Follow.

    1. The allotted subset of the baskets is taken and frequent Item-sets in the subset using simple randomised algorithm is identified. Considering that algorithm, lower the support threshold from s to ps if each map task to get gets fraction p of the complete feed in file . The result is a set of key-value pairs (F, 1), where F is a frequent item-set from the specimen

    2. Each reduce chore is allocated a set of keys, which are Item-sets. The worth is disregarded, and the reduce job simply produces those Item-sets that come into view one or more times . Thus, the result of the first reduce function is the candidate Item-sets.

    3. The map tasks for the second map function take all the output from the first reduce function (the candidate Item-sets) and a section of the input data file. Each map task counts the number of occurrences of each of the candidate Item-sets among the baskets in the section of the dataset that it was allocated. In this second map function (C, v) is the key pair value set will be the output, and where you can see the following parameters as follows.
       C – It is one of candidate sets.
       v – It is the support for the item-set included in the baskets that were input to the map task.

    4. The reduce tasks take the Item-sets they are provided as keys and aggregate the analogous values. The result is the complete support for each of the Item-sets that the reduce task was provided to handle. Those Item-sets whose sum of values is at least s are frequent in the entire dataset . So the reduce task outputs these Item-sets with their sum up . Item-sets that do not have total support at least s are not broadcasted to the output of the reduce task.

Transaction ID	Itemsets
1	A,B,D
2	A,C,E
3	A,D
4	B,E
5	A,C,D
6	A,B,D,E
7	B,D,E
8	B,D

Itemsets	Support
{A}	5
{B}	5
{C}	2
{D}	6
{E}	4

- Now you have to Implement Above SON Algorithm and find the frequent Authors.
- You have to Clean the data in Mapper File.
- Finally you have to use the last reduce file generated and make a flask api to display and fetch the data. And Show The Threshold, Frequent Item-sets or any-other detail possible on an HTML page using flask api.

Submission

You are required to submit a report with response, what you learned by all the group member individually. You will submit 2 Mappers and 2 Reducers. You will also submit Flask Api and HTML Display page.

References :

https://www.geeksforgeeks.org/the-son-algorithm-and-map-reduce/
https://github.com/mongodb/mongo-hadoop/wiki/Streaming-Usage
https://www.youtube.com/watch?v=uhbkyleco2U

EFFORTS DON'T BETRAY YOU

GoodLuck 😊