Forum - Student Groups

Course: 大数据软件工程

Exercise type: Hadoop Python Streaming

Soft Due: 2015-06-01 18:00

Hard Due: 2015-06-01 18:00

Note: hard due has passed. You can no longer submit answer.

Author: 王欣明(TEACHER)

Email: wangxm35@mail.sysu.edu.cn

你已经接近或者达到满分。在完成这道题后，如果愿意的话，请你在下面**Feedback**那里评价一下这道题。你的反馈将同时用邮件发送给作者。

Description:

0 Questions and 0 Answers   Ask New Question

In this project you will work with discussion forum (also sometimes called discussion board) data. It is one type of user generated content that you can find all around the web. Most popular websites have some kind of a forum, and the things you will do in this project can transfer to other similar projects.

This particular dataset is taken from an online forum similar to the popular StackOverflow forum. The basic structure is - the forum has nodes. All nodes have a body and author_id. Top level nodes are called questions, and will also have a title and tags. Questions can have answers. Both questions and answers can have comments. If you are not sure how that all looks, please go to StackOverflow and look around!

You shall run the code mostly on your VMs. The dataset is in the file forum_data.tar.gz. To unarchive it, download it to your VM, put in the data directory and run:

tar zxvf forum_data.tar.gz

There are 2 files in the dataset. The first is "forum_nodes.tsv", and that contains all forum questions and answers in one table. It was exported from the RDBMS by using tab as a separator, and enclosing all fields in double quotes. You can find the field names in the first line of the file "forum_node.tsv". The ones that are the most relevant to the task are:

- "id": id of the node
- "title": title of the node. in case "node_type" is "answer" or "comment", this field will be empty
- "tagnames": space separated list of tags
- "author_id": id of the author
- "body": content of the post
- "node_type": type of the node, either "question", "answer" or "comment"
- "parent_id": node under which the post is located, will be empty for "questions"
- "abs_parent_id": top node where the post is located
- "added_at": date added

The second table is "forum_users.tsv". It contains fields for "user_ptr_id" - the id of the user. "Reputation" - the reputation, or karma of the user, earned when other users upvote their posts, and the number of "gold", "silver" and "bronze" badges earned. The actual database has more fields in this table, like user name nickname, bio (if set) etc, but we have removed this information here.

We might want to help students form study groups. But first we want to see if there are already students on forums that communicate a lot between themselves.

As the first step for this analysis we have been tasked with writing a MapReduce program that for each forum thread (that is a question node with all its answers and comments) would give us a list of students that have posted there - either asked the question, answered a question or added a comment. If a student posted to that thread several times, they should be added to that list several times as well, to indicate intensity of communication.

To make sure your code is running properly, we have put together a smaller data set and set of expected outputs for you to use to check your work. The name of the test data set is student_test_posts.csv.

Run the following command to display your code’s output: $ cat student_test_posts.csv | python mapper.py | sort | python reducer.py

Below you will find the output expected for this exercise when using the test data provided. The output of your code should include all of the rows below, aside from the columns headers, but the order of the rows may be switched around.

Question Node ID |	Student IDs
111	[100000066]
15084	[100004819]

2	[100000005]
262	[100004819]
26454	[100003192]
3778	[100000066, 100008254]
6011204	[100010128, 100020526, 100071170]
6011936	[100004819, 100019875, 100071170]
6012754	[100004819, 100012200]
6015491	[100004467, 100005156, 100071170]
66193	[100002460, 100004467, 100007808, 100071170]
7185	[100003268]

Hint:

Hint is not available for this exercise.

Time limit:

Hard time limitation: 60 seconds

Standard answer spent time: 28 seconds

Your answer:

Already pass (hard) due date: June 1, 2015, 6 p.m. You can no longer submit your code to smart_programmer by yourself.

We don't accept late submission in principle. But if you really have a convincing reason, send the answer code to your TA by email and ask her/him to submit it for you. Don't copy the standard answer. It will be considered as plagirism.

If you want to study the standard answer, you can add it as an optional assignment.

Add this exercise as an optional assignment

Hard due has passed. The standard answer is unlocked:

[*]mapper1.py | combiner1.py | [*]reducer1.py | mapper2.py | combiner2.py | reducer2.py | mapper3.py | combiner3.py | reducer3.py

```
#!/usr/bin/python
import sys

title1 = 'id\ttitle\ttagnames\tauthor_id\tbody\tnode_type\tparent_id\tabs_parent_id\tadded_at\tscore\tstate_string\tlast_
       = '\"id\"\t\"title\"\t\"tagnames\"\t\"author_id\"\t\"body\"\t\"node_type\"\t\"parent_id\"\t\"abs_parent_id\"\t\"ad

def         (    ):
    if str_.startswith("\"") and str_.endswith("\""):
        return      [1:-1]
    else:
        return

def      (    ):
    stuInfo = line.replace("\n"," ").split("\t")
    if    (       ) == 19:
        nodeId = getValue(stuInfo[0])
                 =         (       [7])
        authorID = getValue(stuInfo[3])
                 =         (       [5])
        if nodeType in ["question", "answer", "comment"]:
            if          == "question":
                print "{0}\t{1}".format(nodeId, authorID)
            elif          == "answer":
                print "{0}\t{1}".format(parentId, authorID)
            elif          =="comment":
                print "{0}\t{1}".format(parentId, authorID)

currLine = None
for      in    .     :
    if line == title1 or line == title2:
        continue

          =     .     ().     ("\t")
    if len(items) > 4 and getValue(items[0]).isdigit() and getValue(items[3]).isdigit():
        if          != None:
            mapTo(currLine)
                     =
    else:
        if          == None:
            currLine = line
        else:
            currLine += line

if currLine != None:
         (        )
```

Random test input generator ():

```
```

Discussion:

Discussion is not available for this exercise.

Latest Submission Grade: 100 (submitted on 2015-06-01 12:53)

```
=================================[Phase1: check_plagirism_cloudera]=======================================
Pass.

=================================[Phase2: cloudera_prepare]===============================================
Pass syntax checking. You got 10 points.

=================================[Phase3: cloudera_hadoop_py_streaming_local]=============================
Pass all local test cases. Good job! You get 20 out of totally 20 points.

=================================[Phase4: cloudera_hadoop_py_streaming_cluster]===========================
Spent time: 42.4479532242 seconds. The standard execution time is 28 seconds.
Your hadoop program runs roughly the same fast as the standard answer. You get 100% of the correctness poi

Correctness points: 70
```

Your submission record:

Submission Date	Grade	Diff
2015-06-01 12:53	100	Diff
2015-05-31 14:32	16	Diff
2015-05-31 14:21	30	Diff
2015-05-31 14:03	30	Diff
2015-05-31 12:34	30	Diff
2015-05-31 10:23	30	Diff
2015-05-30 18:42	30	Diff
2015-05-30 17:51	16	Diff
2015-05-30 11:18	10	Diff
2015-05-29 11:07	10	Diff

Top performers in the class:

Name	Points	Total Trials	Last Submission
彭翌	100	11	2015-06-01 01:46
test	100	4	2015-06-01 02:01
吴伟豪	100	12	2015-06-01 03:28
DIS	100	10	2015-06-01 12:53
持盾卫士	100	5	2015-06-01 13:12
把我飘准的普通发凡我	100	7	2015-06-01 13:38
乙同学	100	4	2015-06-01 13:39
我是大傻逼	100	2	2015-06-01 13:41
温伟力	100	8	2015-06-01 13:44
KioKo	100	3	2015-06-01 14:25

Feedback: (这里不是填答案的地方！)

Warning: Don't put your answer into the feedback, as they will be seen by all other students. If you have doubts on your answer, send an email to the author of this assignment instead.

为减轻服务器负担，在评论中插入大图像时建议用链接，小图像的时候建议用base64编码直接嵌入，即打开html编辑页面插入像<img src="data:image/png;base64,iVBORw0KGgoAAAANSclgAAAAEAAA">一样的标签 (图片转base64编码的工具)。注意，评论的大小不能超过100k字节，如果超出就会显示Invalid form data。

难度(选填): ------------------------------

题意表达是否清楚(选填): ------------------------------

对学习的帮助(选填): ------------------------------

你对本题的评价(选填):

Font Family Font Size Paragraph

Path: p

Upload

Feedbacks from students:

Class	Comments

Please contact **Dr. Wang (roadlit at gmail.com)** to report bugs.

Thanks to:

- 吴浩坚同学 and 梁展瑞同学
- Django, Gunicorn, TinyMCE, Sandbox, Nginx
- Valgrind, Google Gode Style SOClone

Recent messages:

- [06/01 14:50] From 王欣明: 最近一段时间经常发生有些同学的作业卡队列的现象。如果发现请及时通知我重启服务队列。
- [06/01 14:49] From 王欣明: 本周是大数据软件工程的期末课程项目，题目会在周二晚上6点开放，同学们可以自己在宿舍做，一周时间内完成。
- [05/31 00:38] From 谢议尊: test

Send message to an user with real name or nick name:

Name:

Message:

Send Message