Lab2 Post MidSem

Date - 13/04/2018

Problem Statement

You are given a new dataset. You have to create a neo4j database and insert the new data into your database. Once your database is loaded with the given data, you have to perform the search queries corresponding to your roll number.

Please find the question numbers corresponding to your roll number from the list provided below and attempt only those questions. Also, go through all the instructions carefully. **Keep in mind the ** points (page 3) while processing your queries**.

1. Given a user mention (author_screen_name), find all the other users(author_screen_names) who are co-mentioned with the given user. Two users are said to be co-mentioned if they are mentioned in the same tweet (tweet_type : Tweet). Sort the output in decreasing order of co-mention count. Your output should only include the following:

   User1 – User2 – Tweet-ids where these users are co-mentioned --Co-mention count

2. Given a location, find the top 3 co-occurring hashtags pairs co-occurring in tweets posted from this location. Two hashtags are said to be co-occurring if they co-occur in the same tweet (tweet_type : Tweet). Sort the output in decreasing order of co-occurrence count. Your output should include only the following:

   Location -- Hashtag1 – Hashtag2 – Tweet-ids where these hashtags are co-occurring -- Co-occurrence count

3. Given a location, find the top 3 co-occurring user-mention (author_screen_name) pairs co-occurring in tweets posted from this location. Two user-mentions are said to be co-occurring if they co-occur in the same tweet (tweet_type : Tweet). Sort the output in decreasing order of co-occurrence count. Your output should include only the following:

   Location -- User-mention1 – User-mention2 – Tweet-ids where these user-mentions co-occur -- Co-occurrence count

4. Given a hashtag, find the top 3 other hashtags co-occurring with this hashtag. Two hashtags are said to be co-occurring if they co-occur in the same tweet (tweet_type : Tweet). Sort the output in decreasing order of co-occurrence count. Your output should include the following:

   Hashtag1 – Hashtag2 – Tweet-ids where these hashtags co-occur – Co-occurrence count

5. Given a location, find the top 2 users (author_screen_names) who are tweeting from that location. Sort them in decreasing order of their tweet count. Your output should include the following:

   Location – user posting a tweet from this location – Number of tweets by this user

6. Given a user mention (author_screen_name), find the top two users (author_screen_names) who mentioned this user. Sort them in decreasing order of mention count. Your output should include the following:

   User-mention – user mentioning user-mention – Tweet-ids of the tweets containing the user-mention -- Mention Count

7. Given the hashtag**,** retrieve the top 3 user-mentions (author_screen_names) that co-occur with this hashtag. A hashtag and a user-mention is said to be co-occurring if they co-occur in the same tweet (tweet_type : Tweet).. Sort the output in decreasing order of co-occurrence count. Your output should include the following:

   Hashtag – User-mention – Tweet-ids where the user mention and the hashtag co-occur – Co-occurrence count

8. Given a hashtag, retrieve top 3 users (author_screen_names) who have used the given hashtag in the tweets (tweet_type : Tweet) that they have posted. Sort them in decreasing order of frequencies. Your output should include the following:

   Hashtag – User – Tweet-ids posted by the user and containing this hashtag –Tweet Count

9. Given a user-mention (author_screen_names), retrieve the top 3 hashtags co-occurring with this user mention. A hashtag and a user mention is said to be

co-occurring if they co-occur in the same tweet(tweet_type : Tweet). Sort the output in decreasing order of co-occurrence count. Your output should include the following:

User-mention – hashtags – Tweet-ids where the user-mention and the hashtag co-occur

10. Given a location, retrieve all user-mentions (author_screen_names) appearing in tweets (tweet_type : Tweet) posted from that location. Sort the user-mentions in decreasing order of frequency. Your output should include only the following.

    Location -- User-mentions -- Tweet-ids posted from this location and containing this user mention – Tweet count

11. Given a user (author_screen_name), find all the other users (author_screen_names) who have replied to the tweets posted by the given user. Display the output on the basis of reply count per tweet. Sort users in decreasing order of the number of replies. Your output should include the following :

    User1 – Tweetid of tweet posted by user1 – User replying to this tweet –Tweet-id of the replies (tweet_type : Reply) -- *Reply Count*

**All string searches should be exact matches. Do not consider substring or case insensitive searches.

**If not mentioned otherwise, all user searches should be based only on the author_screen_name irrespective of the user ids.

**If not mentioned otherwise, while searching for tweets, consider only tweets with type 'tweet'.

**Create uniqueness constraint on hashtag name, author_screen_name and tweet_id. You may use additional constraints as per your judgement and necessity.

Instructions:

1. Attempt only the questions corresponding to your roll number.
2. It is expected that your computer is installed with the required database (Neo4j). You are not allowed to access the database from other machines. If you have not installed Neo4j database in your respective system, you will be excluded from evaluation.
3. Implementation time will be for 90 mins. You will be allowed to enterthe Lab till 9:30AM only.
4. Time allowed for implementation is 90 minutes (9:00 am to 10:30 am). The remaining time is for evaluation.
5. **Input to the queries will be provided before the evaluation.** It will be uploaded 15 mins before the evaluation time.
6. At the time of evaluation, you have to show the output and submit the following files in the presence of the TA:
   a. Export data model from the neo4j console in .png format
   b. Output the results for each question in json format. Since the UI of the graph cannot be used for comparison, you will need to submit the raw output from the Neo4j database, which is submitted as an input to your API for producing the graph.
   c. Program files. The python/php/html codes for all the queries should be clubbed in one program file for each filetype. Do not submit any database files or package default files.
   d. Screen shot of your GUI (each query will need to produce a graph)
   e. Another 15 min will be given to prepare the require file. Altogether, you will get 90+15=105min for programming and preparing the files to be uploaded.
7. Create a single compressed file (.zip or .tar) that will include all the files mentioned in 4. Name it **Assignment_2_<roll number>.zip/tar**. Send it to the following gmail id: cs3452018@gmail.com with subject name as Assignment 2 <roll_no>.
8. Roll number wise allotment of questions is in the following page.
9. Marks Distribution: will be same as before.
10. Please read the instructions carefully and adhere to them.

Additional Information:

If you are using Python programming language and py2neo database package, you can use the .data() method to dump the entire result in a json file

http://py2neo.org/v3/database.html#py2neo.database.Cursor.data

If you are using Php , you can use the json_encode function to dump the entire result into a json file.

https://stackoverflow.com/questions/31997517/export-json-to-file-from-neo4j-databse-using-php