DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

Project Description

CS-322 Introduction to Database Systems
Spring 2020

Table of Contents

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

Introduction

In this project the students will get a set of data files. Based on that data, they will i) design a database schema, ii) parse, clean, and load the data into a DBMS, iii) write queries, and, finally, iv) evaluate and optimize queries with index structures/query rewriting in order to analyze the performance impact on generated query plans and discuss about the query optimizer decisions on querying a given dataset.

IMPORTANT: Read the whole document before starting doing any work.

Short Description of Project

The dataset contains a subset of data from Yelp, a business directory and crowd-sourced review forum. The project is done in teams of 3 people. The project is separated into 3 milestones, which follow the material taught in the lectures. We have synchronized each milestone with the material of the lectures for your convenience.

The first milestone requires you to analyze the dataset and extract the E-R (Entity-Relationship) schema as well as getting acquainted with a DBMS. The second milestone requires you to express a simple set of queries on top of the loaded database. The goal of this part is to familiarize with data loading and the challenging task of data cleaning. You will also get to apply your SQL skills, and get a first intuition about how query performance is directly dependent on i) the way you formulate a query and ii) the logical and physical design of your database. Finally, in the third part of the project you will express a set of more sophisticated SQL queries, which you will also analyze to come up with a detailed description of the execution. You will analyze the queries and their respective query plans in order to optimize the execution, either with building appropriate index structures or rewriting the queries to make the execution more efficient (or both), and discuss about the decisions that query optimizer took, such as if it even considered the newly created index structure.

For each of these milestones the students should prepare a document following the provided template which describes the completed work. The grading will be done based on the final report as well on a presentation and short discussion with the TAs. The final report should contain material about all the work done for the 3 milestones combined into one document**. The reports after the first two milestones are optional, while the final (3rd) deliverable is mandatory and gradable. However, only the teams that submit the intermediate reports will get feedback on their progress.**

IMPORTANT: Only the teams that deliver the intermediate milestone deliverables within deadline will receive feedback!

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

Deliverable 1: Create ER model, Design & Create Schema

Deadline (to get feedback): 23/03/2020

The students will use the data from the following data files:

- yelp_academic_dataset_business.csv
- yelp_academic_dataset_review.csv
- yelp_academic_dataset_tip.csv
- yelp_academic_dataset_user.csv

The goal of this deliverable is to design an ER model and a corresponding relational schema, and create the database tables in a database system. The organization of the data in files and the given description ***DOES NOT IMPLY*** an ER model or a relational schema (e.g. that these are the only 4 entities of the E-R model). It is given to help the student understand the format of the data faster. Finally, a discussion about constraints and removing redundant information should be included in the project report.

In the 1st deliverable the students should:

1. Create an ER model for the provided data. For your ease, you may provide a relational translation of the ER model, that will help you with the next point.
2. Design the database and the constraints needed to maintain the database consistent.
3. Provide the SQL DDL commands to create the tables and the constraints in a relational database system.
4. Describe their work in the form of a report which should contain an ER diagram, SQL DDL code for table creation, description of the data constraints, and justification of the design choices (in a few paragraphs). The report should be submitted as a single pdf file (**one PDF document per group**).

Important Note: Before designing an E-R schema, understand the data and read carefully the notes given in the form of **FAQ** at the end of the project description. If you need any clarifications, ask the TAs during the project session or office hours.

Tip: Analyze the data and keep in mind that a column in CSV file does not always map to a column in entity/table. Remember that the column values have to be atomic (not a list) in relational model (1st Normal Form). Some data columns may become separate tables for this reason. Feel free to group some values/attributes into a separate entity/table if they seem to **repeat** or appear to be logically a separate entity (and then you can explain your assumptions over the data and design decision).

Points breakdown for elements of this deliverable: 18 points for the ER model, 4 points for DDL + constraints.

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

Deliverable 2: Import Data. Basic SQL queries

Deadline (to get feedback): 27/04/2020

In this phase, students have to import the provided raw data into their database. The students should know how to insert/delete/update data via SQL commands, as well as to execute simple exploratory queries over the data.

Students have to implement the following queries in SQL:

1. What is the average review count over all the users?
2. How many businesses are in the provinces of Québec and Alberta?
3. What is the maximum number of categories assigned to a business? Show the business name and the previously described count.
4. How many businesses are labelled as "Dry Cleaners" or “Dry Cleaning”?
5. Find the overall number of reviews for all the businesses that have more than 150 reviews and have at least 2 (any 2) dietary restriction categories.
6. Display the user id and the number of friends of the top 10 users by number of friends. Order the results by the number of users descending (the user with the highest number of friends first). In case there are multiple users with the same number of students, show only top 10.
7. Show the business name, number of stars, and the business review count of the top-5 businesses based on their review count that are currently open in the city of San Diego.
8. Show the state name and the number of businesses for the state with the highest number of businesses.
9. Find the total average of “average star” of elite users, grouped by the year in which they started to be elite users. Display the required averages next to the appropriate years.
10. List the names of the top-10 businesses based on the median “star” rating, that are currently open in the city of New York.
11. Find and show the minimum, maximum, mean, and median number of categories per business. Show the final statistic (4 numbers respectively, aggregated over all the businesses).
12. Find the businesses (show 'name', 'stars', 'review count') in the city of Las Vegas possessing 'valet' parking and open between '19:00' and '23:00' hours on a Friday.

In summary, in the 2nd deliverable the students should:

1. Parse the given data and import them in the created database as described in your 1st deliverable.
2. Implement (using SQL) the queries described above.
    a. Provide the SQL code as well as the first 5 rows (when applicable) of the result for each query.
    b. Note: Consider the use of indexes to accelerate long-running queries.
3. Extend the project report from the first deliverable with the description of the work done for the second deliverable and an explanation for the design choices. Include any changes to the design covered in the first deliverable, with justification of the changes. The report should be submitted as a single PDF file.

Points breakdown for elements of this deliverable: 15 points for the queries, 8 points for data cleaning and loading to the DBMS (4 for parsing and loading, 4 for cleaning and assumptions/explanations).

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

Deliverable 3: Interesting SQL queries

Deadline (you must submit the report to get graded): 29/05/2020

A series of more interesting queries should be implemented with SQL. In addition, performance of **any 5 queries** should be optimized and analyzed in depth by using indexes and evaluated based on the produced query plans.

The queries to be implemented are:

1. What is the total number of businesses in the province of Ontario that have at least 6 reviews and a rating above 4.2?
2. What is the average difference in review scores for businesses that are considered "good for dinner" that have noise levels "loud" or "very loud", compared to ones with noise levels "average" or "quiet"?
3. List the “name”, “star” rating, and “review_count” of the businesses that are tagged as “Irish Pub” and offer “live” music.
4. Find the average number of attribute “useful” of the users whose average rating falls in the following 2 ranges: [2-4), [4-5]. Display separately these results for elite users vs. regular users (4 values total).
5. Find the average rating and number of reviews for all businesses which have at least two categories and more than (or equal to) one parking type.
6. What is the fraction of businesses (of the total number of businesses) that are considered "good for late night meals"?
7. Find the names of the cities where all businesses are closed on Sundays.
8. Find the ids of the businesses that have been reviewed by more than 1030 unique users.
9. Find the top-10 (by the number of stars) businesses (business name, number of stars) in the state of California.
10. Find the top-10 (by number of stars) ids of businesses per state. Show the results per state, in a descending order of number of stars.
11. Find and display all the cities that satisfy the following: each business in the city has at least two reviews.
12. Find the number of businesses for which every user that gave the business a positive tip (containing 'awesome') has also given some business a positive tip within the previous day.
13. Find the maximum number of different businesses any user has ever reviewed.
14. What is the difference between the average useful rating of reviews given by elite and non-elite users?
15. List the name of the businesses that are currently 'open', possess a median star rating of 4.5 or above, considered good for 'brunch', and open on weekends.
16. List the 'name', 'star' rating, and 'review_count' of the top-5 businesses in the city of 'los angeles' based on the average 'star' rating that serve both 'vegetarian' and 'vegan' food and open between '14:00' and '16:00' hours. Note: The average star rating should be computed by taking the mean of 'star' ratings provided in each review of this business.
17. Compute the difference between the average 'star' ratings (use the reviews for each business to compute its average star rating) of businesses considered 'good for dinner' with a (1) "divey" and (2) an "upscale" ambience.

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

18. Find the number of cities that satisfy the following: the city has at least five businesses and each of the top-5 (in terms of number of reviews) businesses in the city has a minimum of 100 reviews.
19. Find the names of the cities that satisfy the following: the combined number of reviews for the top-100 (by reviews) businesses in the city is at least double the combined number of reviews for the rest of the businesses in the city.
20. For each of the top-10 (by the number of reviews) businesses, find the top-3 reviewers by activity among those who reviewed the business. Reviewers by activity are defined and ordered as the users that have the highest numbers of total reviews across all the businesses (the users that review the most).

In total, in the 3rd deliverable the students should:

1. Accommodate all above queries by giving the corresponding SQL code.
    a. Note: Consider the use of indexes to accelerate your long-running queries.
2. Select 5 queries from Deliverable 3, and accelerate them by using indexes. Explain the necessities of indexes based on the queries and the query plans that you can find from the system (you are free to select any 5 queries you like from the queries of the 3rd deliverable).
3. After the introduced optimizations, report the runtime of all queries in milliseconds and explain the distribution of the cost (based again on the plans) for the 5 queries selected in part 2.
4. Present the results of the queries.
6. Complete the project report written for the previous deliverables by adding description of the queries, explanation for the design choices, analysis of the chosen queries, as well as the changes compared to the work described in the previous deliverables. The report should be submitted as a single PDF file.

Points breakdown for elements of this deliverable: 50 points for the queries, 5 points for optimization.

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

Yelp! data description

In this section, we present the data on which the project is based. Read carefully the data description, the FAQ and if in doubt ask the TAs for clarification. The data is stored in CSV (comma separated values) files.

yelp_academic_dataset_business.csv

This file contains information about the businesses listed on Yelp.

1. Address: the provided business address
2. Attributes: a list of business attributes/specializations. The attributes themselves can be further a list having descriptors of an attribute. **Tip:** there are overall about 6 unique groups in the dataset for attributes, that contain some number/a list of further descriptors. Make sure you follow 1st Normal Form when designing your ER model!
3. Business_id: the unique ID of the business
4. Categories: string representing a list of assigned categories. 1000+ possible unique values overall in dataset. Keep this in mind when parsing and designing your ER model!
5. City: name of the city
6. Hours: list of opening/closing hours per day
7. Is_open: indicates if the listed business is currently open for business
8. Latitude: geographical latitude
9. Longitude: geographical longitude
10. Name: listed name of the business
11. Postal_code: postal code of the business
12. Review_count: the aggregated number of reviews (does not have to match the actual ones in data)
13. Stars: the aggregated number of stars (does not have to match the actual ones in data)
14. State: the name of the state where business is located

yelp_academic_dataset_review.csv

This file contains information about the reviews that users leave on businesses.

1. Business_id: the unique ID of the business this review relates to
2. Cool: the number of users that rated this review as cool
3. Date: the date the review was posted
4. Funny: the number of users that rated this review as funny
5. Review_id: the unique ID of the review
6. Stars: the number users that starred this review
7. Text: the text description of the review (shortened)
8. Useful: the number of users that found this review useful
9. User_id: the unique ID of the user that wrote this review

Tip: As this is a CSV, meaning that the values are split by commas (“,”), and new line (“\n”) separates different rows, make sure to properly handle the **Text** field, as it may contain some of these values between its quotes!

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

yelp_academic_dataset_tip.csv

This file contains information about the tips (advice) users give about the businesses.

1. Business_id: the unique ID of the business this tip relates to
2. Compliment_count: the number of users that have complimented this tip
3. Date: the date this tip was posted
4. Text: the text description of the tip (shortened)
5. User_id: the unique ID of the user that wrote this tip

Tip: As this is a CSV, meaning that the values are split by commas (“,”), and new line (“\n”) separates different rows, make sure to properly handle the **Text** field, as it may contain some of these values between its quotes!

yelp_academic_dataset_user.csv

This file contains information about the users.

1. Average_stars: average stars this user has received from other users for his reviews
2. Compliment_cool: the number of cool compliments this user received
3. Compliment_cute: the number of cute compliments this user received
4. Compliment_funny: the number of funny compliments this user received
5. Compliment_hot: the number of hot compliments this user received
6. Compliment_list: the number of list compliments this user received
7. Compliment_more: the number of more compliments this user received
8. Compliment_note: the number of note compliments this user received
9. Compliment_photos: the number of photos compliments this user received
10. Compliment_plain: the number of plain compliments this user received
11. Compliment_profile: the number of profile compliments this user received
12. Compliment_writer: the number of write compliments this user received
13. Cool: the number of cool votes sent by this user
14. Elite: list of the years in which this user has an elite status
15. Fans: the number of fans this user has
16. Friends: list of friends, whose elements are the user_id of the friends (**who are also users on Yelp**).
17. Funny: the number of funny votes sent by the user
18. Name: user’s first name
19. Review_count: the number of reviews this user has written
20. Useful: the number of useful votes sent by the user
21. User_id: the unique ID of the user
22. Yelping_since: the date when the user joined Yelp

You can find the data here: http://diaswww.epfl.ch/courses/cs322/2020/project/yelp.zip.

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

Frequently Asked Questions

How does one browse the data?

The dataset size is substantial, so it is hard to open most files using a notepad or text editor.
Applications such as Notepad++ and Sublime Text do a better job, but may still have issues with bigger files.
We thus also propose using Unix commands such as:

1. head: prints the first 50 lines of the file
2. less: allows backward movement in the file as well as forward movement
3. vi text editor: this editor does not open the whole file but only the part that is displayed

A useful and recommended method is to browse the data using scripting languages such as Python, where you can use Pandas library to load the CSV as a DataFrame, and explore parts of data via the functions of the library. This way it is also useful to explore the data for future data cleaning, transformation, and loading to DBMS, and the library also provides a method to explore basic statistics and features of the data.

Which is the format of the given data?

The given data is CSV files (Comma Separated Values) which are values separated with comma (,). Each column represents a specific attribute. Usually in CSV files the name of the attribute is given in the first line of the file.

Why are the datasets "dirty"?

Real-world data is almost always dirty; missing values are commonplace; users abuse DBMS datatypes and store values based on their arbitrary, ad-hoc rules. We consider data cleaning to be a part of your project. Regarding how to perform data cleaning, there is more than one correct solution. Some possible ways are the following:

- Use your favorite scripting language, or a typical program that handles data inconsistencies. For example, Python, Java, and Scala all feature CSV parsers which you can use to read and transform the data.
    - One proposed way is to use libraries such as Pandas for Python for fast data manipulation, then you can use the same library to save the DataFrame structure to CSV suitable for DBMS loading.
- Use Unix commands such as sed, grep, awk.
- Load the data in a DBMS and then use DBMS functions to transform them based on your requirements.

Which database system should I use?

You are free to use any DBMS you want. Typical open-source examples are MySQL and PostgreSQL. We will also grant you access to an Oracle installation located on a server of the DIAS lab, which you can use. We will do our best to troubleshoot any issues with the Oracle server and help with issues related to your own installations.

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

Which character encoding should I set?

All files use UTF-8 encoding. Take care of initializing your database using the correct encoding before creating tables or loading the data.

What should I do if it takes too long to load the data?

The two most common reasons for a slow data loading process are the following:

1. Defining too many indexes/foreign key relations in your tables can delay loading significantly. **We therefore propose that you first create simple tables with only primary key properties, or without any constraints specified at all**. Once data is loaded, add the more complex table relations and indexes.
2. If you are using the database system provided by us, make sure that you are connected to the EPFL network via cable (i.e., use a machine from the laboratory). If you connect via Wi-Fi or from home via VPN, it may take a long time to upload the data files, thus leading to longer loading times.

An additional scenario is that you have set up your own database server (e.g., PostgreSQL or MySQL), and that the default resources allocated to it are very few. In that case, some useful links are the following:

- https://dev.mysql.com/doc/refman/5.5/en/innodb-buffer-pool.html
- http://www.rathishkumar.in/2017/01/how-to-allocate-innodb-buffer-pool-size-in-mysql.html
- https://wiki.postgresql.org/wiki/Tuning_Your_PostgreSQL_Server

What should I pay attention to?

1. **There is no intermediate grading**
    a. We still urge you to complete the milestones on time, so that you will not be overwhelmed at the end of the semester.
    b. The parts of the project are created in a way so that you will use the things you learn in the course and the exercise session and have hands-on experience.
    c. Every one of your deliverables should include the text from the previous ones too, explicitly updated to reflect the changes you made to address the feedback we gave you.
2. **Collaboration**
    a. We want you to collaborate
    b. We DO NOT GO INTO how you will split the work -> As long as you do equal parts of the work
    c. Writing the queries can (and should!) be done by everyone!
        i. You can solve the queries in multiple ways to find the optimal one (helpful with optimizing)
3. The only important deadline on which you are graded is the last one **BUT** if you want feedback make sure to send us the milestone deliverables!

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

Can I discard some data?

Dropping some erroneous values is acceptable. Under no circumstances, however, should you drop a significant chunk of the data. Whenever you drop some data, you should include the description of the dropped data and the reason for doing so.

How long should the deliverables be?

There is no strict page limit, as long as the deliverables report on the points we requested and are informative.

How should I choose my team?

Putting teams together is entirely up to you. Our advice is that every team member should be exposed equally to every task of the project. While, for example, it might appear tempting to a good data analyst to focus on the data cleaning and loading and quickly finish her assigned task, she will then be disadvantaged in the course midterm and final, because her SQL and query optimization experience will be limited.

What should I do if one of my teammates does not work?

We advise that you address the issue early on, before you encounter high load due to a deadline. We cannot be more lenient to such teams as a whole for fairness reasons. During the final project presentation, however, it becomes obvious whether a team member did not place equal effort; this student will get a lower grade.

When can I ask questions about the project?

The weekly project session is the intended place for questions. Otherwise, please use the Moodle forum for questions that are of interest to your colleagues, too. Finally, every TA has specified office hours that you can use for further clarifications.