Yelp Data Extraction and Exploration

This work is completed by the following three people. Everyone is assigned an even amount of work.

Name	NetID	Department	Degree Program
Luoqi Wu	lw31	Computer Science	Bachelor
Yan Xu	yx28	Chemical and Biomolecular Engineering	PhD
Zheng You	zy24	Computer Science	Master

1 High-Level Statement

The goal of this project is to extract data from the Yelp dataset and to perform data explorations. The overall project can be divided into five phases: data import, data comprehension, data cleaning and profiling, Entity-Relationship Diagram creation, and data exploration. The following tasks are categorized according to the five phases.

2 Task Assignment

2.1 Data Import (Zheng You)

1. Conduct researches about loading JSON formatted data into PostgreSQL.

   There are multiple ways to load the JSON formatted data. One way is to convert the JSON data into CSV format and load the CSV data into PostgreSQL. Another way is to operate JSON data directly with PostgreSQL because it supports native JSON data type since version 9.2 [3].

2. Import the JSON formatted data into PostgreSQL to create relations based on a chosen method.

   There are a variety of business types in the raw dataset and we are interested only in the restaurant type. It would be appropriate to filter out non-restaurant type business in the data import process. This would reduce the data imported in PostgreSQL.

2.2 Data Comprehension (Yan Xu)

1. Identify a list of interesting questions from both customer and business perspectives.

   We have identified three types of key questions:
   - business and review data exploration;
   - user and review data exploration;
   - restaurant recommendation.

   It should be noted that a key question will be decomposed into multiple sub-questions. The detailed sub-questions are listed in Section 2.5.

2. Understand relations and identify fields of interest.

   The fields of interest are chosen based on the key questions. For instance, attributes "categories", "city", "state", "business_id" from "business" relation are needed to answer some of the sub-questions from the first key question.

2.3 Data Cleaning and Profiling (Luoqi Wu)

1. Clean the raw data.

   Before we dive into profiling the data and extracting derived relations, we need to perform some clean-up. This process is mainly based on the understanding we have developed on the dataset. For example, suppose we identify the "text" field in the "yelp_academic_dataset_review.json" file as relevant, and also want to limit its character length within 5000, the Yelp max character length. We could clean up the file by adding a column "text_len" that contains the length of the review body text. Based on this column, we can easily filter out reviews that do not fit our standard. Similar procedures will be performed for other fields of interest.

2. Profile the data.

   We will mainly perform three types of data profiling: single column profiling, cross-column analysis, and inter-table analysis. Single column profiling will focus on improving understanding of the frequency distribution of different values, types, and use of each column (e.g. average length of Yelp reviews). Embedded value dependencies can be exposed in a cross-columns analysis (e.g. relationship between a business's average rating and its location). Finally, overlapping value sets possibly representing foreign key relationships between entities can be explored in an inter-table analysis (e.g. the set of businesses that a user has reviewed) [1].

2.4 Entity-Relationship Diagram Creation (All team members)

1. Identify the entities and relationships.

   The entities consist of the imported raw relations and derived relations. Referential integrity will be enforced with foreign keys. The Yelp dataset has already provides Ids for each relation, which can used as primary keys.

2. Draw the Entity-Relationship diagram based on defined entities and relationships.

   There are mainly three types of entities in the Entity-Relationship diagram: business, user, and review. There will be multiple derived entities associated with each type to facilitate queries. For instance, the third key question restaurant recommendation requires frequent uses of categories. It would a wise choice to create a relation to contain the category for each business.

   The assignment of entity and relationship design is as follows.
   - Business entity and associated derived entities (Zheng You)
   - User entity and associated derived entities (Luoqi Wu)
   - Review entity and associated derived entities (Yan Xu)

2.5 Data Explorations (All team members)

2.5.1 Business and Review Data Exploration (Luoqi Wu, Yan Xu)

1. What are the top 5 restaurant categories with highest ratings in each city?

   The result is used to form the derived relation "high_ rated_category_per_city", which will be used later. The category information is taken from the derived relation "business_category". The rating information is taken from the "stars" attribute in the "business" relation.

2. What are the average rating based on business profiles and the average rating of every year based on reviews for restaurant categories in each city?

   We are interested to see the rating fluctuations with time for restaurant categories in each city. To ensure there are enough reviews, we will consider categories only with at least a certain amount of reviews.

2.5.2 Review and User Data Exploration (Yan Xu, Zheng You)

Fraudulent user identification: we try to find fraudulent users by comparing the user ratings towards restaurants and the actual restaurant ratings. This is assuming the attribute "stars" in the "business" relation is accurate.

1. Calculate a user's rating towards each restaurant this user has visited based on reviews.

2. Calculate the difference between the user's rating and the actual rating of each restaurant for each user. Compute the mean and variance of these differences for each user.

3. Identify fraudulent users.

Variance \ Mean	Low	High
Low	Normal	Fraudulent
High	Suspected	Fraudulent

   We define users with high difference mean as fraudulent regardless of variance values. Users with low difference mean but high difference variance are suspected to be fraudulent. We will use the received compliment count in the user profile to further clarify the suspected users. Those with high compliment count are not fraudulent.

4. Calculate the ratio of fraudulent users to users who have written reviews. Compare this ratio with some sources from Yelp.

Gender-rating relationship: we would like to find the relationship between genders and ratings.

1. Add an additional column "gender" to the "user" relation.

   The raw dataset does not include the gender information in the "user" relation. We will try to determine the user gender by the name through common male and female names. It is not a very accurate estimate but we believe it is sufficient for our purpose.

2. Calculate the average review count and average rating for male and female in total.

   From this query result, we would see the relationship between genders and ratings.

3. Calculate the average review count and average rating for male and female for top 10 restaurants with most reviews.

   This would provide some suggestions to these restaurants on attracting certain gender customers.

2.5.3 Restaurant Recommendation (Luoqi Wu, Zheng You)

In this key question, we try to compare different restaurant recommendation strategies: recommendation by city and recommendation by friend.

1. Identify personal preferred restaurant categories, called set **U**.

   This can be identified by calculating the review count and the average rating a user gives for each category. We will define a metric to take these two factors into consideration. The set **U** consists of the top 5 categories with the highest metric.

2. Identify restaurant categories recommended by city.

   (a) Identify the user city.
   We define the user city as the city a user visited most. The city information is only included in the "business" relation. We intend to use the "review" and the "business" relations to find the city which a user frequents most.

   (b) Find the recommended restaurant categories based on the user city.
   In the first key question, we have found and created the relation "high_rated_category_per_city". We would recommend those categories to users.

   (c) Calculate the Jaccard index[2] between a user's preference set **U** and the city recommendation set **C**, where

$$\text{Jaccard Index} = \frac{|U \cap C|}{|U \cup C|}$$

   The Jaccard Index provides a similarity measure over sets.

   (d) Aggregate the Jaccard Index over multiple randomly chosen users to compute the Robustness Index, where

$$\text{Robustness Index} = \frac{1}{n}\sum_{i=1}^{n}(\text{Jaccard Index}_i)$$

   The higher this index is, the better the recommendation by city strategy is.

It should be noted that step 2a to 2c will be applied for every random chosen user. So it would be convenient to implement these steps as imperative SQL programs.

3. Identify restaurant categories recommended by friends.

   (a) Identify the user friend cycle.
   This is a connected component problem and we will employ an imperative SQL program to implement breath-first search (bfs) algorithm to solve this problem.

   (b) Find recommended categories from the user's friend cycle, called set **F**.
   When a user's friend cycle and each user's personal preference are determined, the recommended categories are the top 5 which are most common in the group.

   (c) Calculate the Jaccard Index between a user's preferences set **U** with the friend cycle recommendation set **F**.
   $$\text{Jaccard Index} = \frac{|U \cap F|}{|U \cup F|}$$
   The Jaccard Index provides a similarity measure over sets.

   (d) Aggregate the Jaccard Index over multiple randomly chosen users to compute the Robustness Index, where
   $$\text{Robustness Index} = \frac{1}{n}\sum_{i=1}^{n}(\text{Jaccard Index}_i)$$
   The higher this index is, the better the recommendation by friend cycle strategy is.

   It should be noted that step 3a to 3c will be applied for every random chosen user. So it would be convenient to implement these steps as imperative SQL programs.

4. Compare the Robustness Indices between the two strategies and determine which one provides a better recommendation. This provides a suggestion for the Yelp recommendation strategy.

We believe the work is balanced among team members and is distributed by each one's advantages. Zheng and Luoqi are major in Computer Science who have relatively solid programming backgrounds. Hence the data import and processing are assigned to them. Yan is a PhD candidate with strong analytical ability who will bring insights to the data. In addition, all team members will participate in the Entity-Relationship diagram creation. The final data exploration tasks is distributed evenly through key questions.

Reference

[1] *Data Profiling*, URL: `https://en.wikipedia.org/wiki/Data_profiling`.

[2] *Jaccard index*, URL: `https://en.wikipedia.org/wiki/Jaccard_index`.

[3] *PostgreSQL JSON*, URL: `http://www.postgresqltutorial.com/postgresql-json/`.