Yelp Dataset Analysis

Team 2: Mahima Manik[1] and Sruti Goyal[2]
[1] 2017MCS2093, M.Tech Computer Science, IIT Delhi
[2] 2017MCS2078, M.Tech Computer Science, IIT Delhi

***Abstract* – The purpose of review-based online platforms is to facilitate the choice of users for the services they are interested in or want to avail. When we have such large dataset, then the interest of the users towards a product/service can be understood from the reviews and thus can be used to improve the product/service. In this coursework, we aim to extract such information for the benefit of businesses.**

1) INTRODUCTION

Yelp was founded in October, 2004 by Jeremy Stoppelman and Russel Simmons. The initial idea of Yelp was an email based referral network to help people find great businesses around them. Their initial implementation was so convoluted that it got rejected by the investors. Usage data showed that users were not answering requests for referrals, but were using the " Real Reviews" feature, which allowed them to write reviews unsolicited. In 2005, they redesigned their website after which they were able to raise $ 15 million funding. Number of reviewers also increased considerably. Yelp had a monthly average of 28 million unique visitors who visited Yelp via the Yelp app and 74 million unique visitors who visited Yelp via mobile web in Q2 2017. Yelpers have written more than 135 million reviews by the end of Q2 2017. According to *BusinessWeek*, Yelp has complicated relationship with small businesses. It trains small businesses in how to respond to reviews, host social events for reviewers and provides data about businesses.

2) DATA COLLECTION

The data collected from Yelp is in JSON format to apply transformation on dataset using Spark. The data available to us is about various business and users registered on Yelp. Yelp acts as a link between businesses and users. Users can provide their reviews, ratings, tips, photos, etc. based on their experience with a particular business. Yelp dataset contains the following data:

Business: Each business is identified by a unique id. The dataset contains business id, name, address, number of reviews it has received, stars, opening hours of each day in a week, category of a business and various attributes of a business. There are 1,56,639 businesses registered with Yelp.

Users: Each user is assigned a unique id. The dataset contains user id, name, the number of reviews written by a user, the duration from when a user is registered with Yelp, the number of tags(useful, funny or cool) given by the user to other reviews, the number of fans the user has, the number of various compliments that a user has received, the average stars received on the reviews given by the user, a list of friends the user has on Yelp and the list of years the user was with Yelp Elite Squad(YES). The number of users registered with Yelp is 11,83,362.

Review: Each review is assigned a unique id. The dataset contains for each review, its id, the review text, the date at which the review was posted, the user id of the Yelper who posted it, the business id for which the review was written, the stars that a review received from other Yelpers and the number of tags(funny, useful or cool) recieved on the review.

Checkin: The dataset contains for each business id, the number of checkins at different times for each day of the week.

Tip: Tips are shorter than reviews and tend to convey quick suggestion. The dataset contains the tip text, the user id who has posted the tip, the business id for which the tip was posted, the date on which it was posted and the number of likes it has received.

3) PROBLEM STATEMENT

The Yelp dataset is mainly focussed on reviews. This helps users to have a great number of choices among the businesses and choose the best suited to them. When new businesses open up, Yelp dataset can help them to spot the right opportunity at the right place. Thus our analysis was focussed on how businesses are benefitted from the dataset. The information that was drawn out from the reviews can be made beneficial for the new businesses who want to join the market. Even the existing businesses can benefit from it by drawing analysis such as what can be improvised or which services can be added to help them stand out in the market. This way they can increase their revenues and also increase their Yelp ratings.

4) NEED FOR ANALYSIS

Yelp has thousands of businesses in the same domain in any particular geographical area. Each business needs to know how well it is performing relative to the other businesses in the same domain. Such relative performance of businesses helps them to work, in order to improve their ranking, as well as revenue and their position in the market. This also helps customers to decide, which business is best for them in the area they are looking for. Also, before setting up new business in an area, it is necessary to analyze the scope of that business in the target area. Our analysis will tell how many businesses are already there in the target area and how well they are doing.

5) METHODOLOGY

For the given problem statement, it requires the reviews of the users. Reviews are text-based and they can be positive, negative or neutral. Ranking is given to each business based on the reviews received by them. The comparisons between the businesses in the same domain is done on the basis of their ranking. We then plot businesses of the same domain with their rankings in a given geographical area on the map for visualization purpose.

First we explored the data and pre-processed it to remove any ambiguity. After pre-processing the data, we applied various queries on the dataset.

We partitioned the data as per location and their categories, and formed smaller datasets with names " <category>_ <location>.json" , e.g., food_ NorthYork.json, beauty_ Edinburgh.json.

We collected reviews of each business and stored them in separate files for ease of analysis and named the files as " rev<business_ id>.json" , e.g., rev_ JO4bxQ9hRX-42XFMlpyKA.json.

For each business in a particular category and city, we found out how many reviews were positive, negative and neutral. These factors were then used to find the ranking of the business.

The rating was calculated using Bayesian rating:

$$r = \frac{(AvgVotes * AvgRate) + (ThisVotes * ThisRate)}{AvgVotes + ThisVotes}$$

where,

AvgVotes: Average number of reviews all the businesses have received(including only those who have reviews>=1)

AvgRate: Average number of positive reviews for all the businesses

ThisVotes: Total number of reviews on this business

ThisRate: Average number of positive reviews on this business

We rated the businesses based on the types of reviews a business has received. We then generated rank on each business (rank_ positive).

6) IMPLEMENTATION

Apache Spark is a general-purpose engine for large-scale data processing. We configured Spark in standalone mode for this purpose. We configured each of our virtual machines in the following way:

SPARK_ WORKER_ INSTANCES=4

SPARK_ WORKER_ MEMORY=4g

One of the nodes was made the Master node and the other three node were slaves.

Spark provides efficient in-memory computations for large data sets by distributing computation across multiple computers. We used Dataframe API of Spark. The DataFrame is a collection of distributed Row types. These provide a flexible interface and are similar in concept to the DataFrames in Python (pandas) as well as in the R language.

Dataframes were used to store our files, because they can be easily distributed over all the nodes in the cluster. Input files like business.json, review.json and other intermediate results were handled as dataframes. Dataframes were located across machines in the cluster. The source file was stored at each of the nodes.

We used Tableau for the visualisation purpose. We exploited the geographical data available to us such as latitude, longitude and city. We plotted businesses and their calculated rankings on the world map for visualisation.

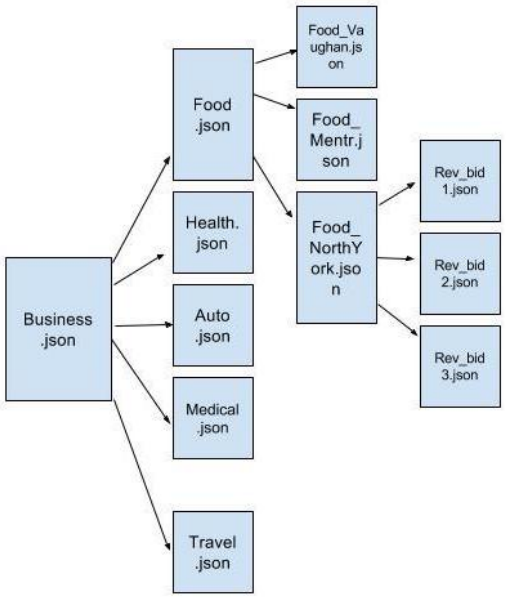

Figure 1: File Hierarchy

7) RESULTS AND INFERENCES

Yelp is actively used in more than thousand cities across the world. In each city, there are businesses of various categories. Businesses are ranked category wise in the area.

The business.json file contains the categories associated with each business. We found out all the distinct categories from this file and partitioned business.json into Spark Dataframes for each such category. Now each category file was partitioned as per the city it is located in. We obtained all businesses of the same category in the same city. Furthermore, reviews of each business in the given category was found out, by partitioning reviews.json. It was then analyzed and rank was decided for each business.

For instance, Figure 2 shows the visualization of businesses that fall under the category of Health & Medicines in the city Madison. The red dots show various health businesses registered with Yelp in Madison. The numbers with each of the red dots indicate the rankings that our analysis has generated. An upcoming health business, for example, can figure out from the map that if it opens up in the upper right corner of the map, growth opportunities will be less compared to opening in upper left corner. This can be inferred because the businesses in the upper right corner have ranks 4, 11, 14 and 18 as compared to businesses on upper left with ranks 43, 118 and 33.

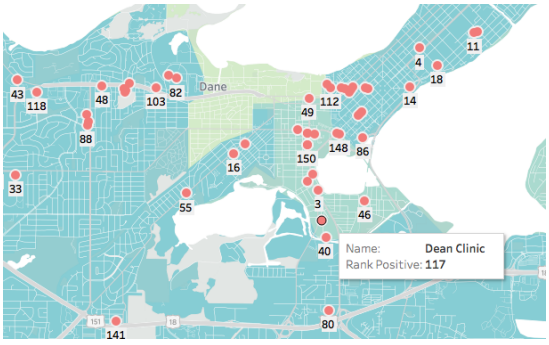

Figure 2: Health businesses in Madison

Running a query approximate 200 times on 4GB file using simple file structure, took approximate 5 hours, on 16GB RAM. Running the same program on Spark (4GB cluster), using dataframes took 2:30hours. It further got reduced when 3 slaves were employed to 70 minutes. The figure below shows this comparisons on different configurations.

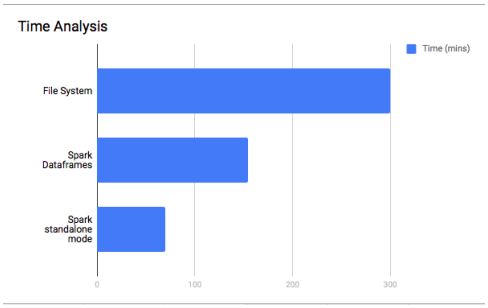

Figure 3: Time Analysis

Users can also be benefitted by comparing the ranks of various businesses on the map.

8) REFERENCES

a) Yelp Dataset: https://www.yelp.com/dataset/

b) Yelp Wikipedia: https://en.wikipedia.org/wiki/Yelp

c) TheBroth.com: http://thebroth.com/blog/118/bayesian-rating.html

d) Fulmicoton.com: https://fulmicoton.com/posts/bayesian$_$rating

e) Apache Spark: https://spark.apache.org/docs/2.2.0

f) Python TextBlob: https://textblob.readthedocs.io/en/dev