Introduction

With more and more access and ease in using the Internet, an increasing number of people are writing reviews for their experiences (Hu, Liu, 2004). As a result, the number of reviews that a product receives grows rapidly over the recent years, providing cascades of complex information. As humans are innately pleased by good food and unique dining experiences, Yelp nowadays became the most popular online consumer review website used for local business reviews and recommendations (Bird, 2015). The abundance of Yelp reviews provides sufficient starting point for doing text analysis. Building on past users' review, in this project, we will take on a creative approach to do restaurant recommendation to users, by performing a detailed text analysis of users' reviews on Yelp.

Data

This year, Yelp presents the ninth round of its own dataset challenge. Yelp released a dataset that includes information about local businesses in 11 cities across four countries. All data were enclosed in five *json* files. For our own convenience, we converted all the *json* files to *csv* format files. In our project, we used three sub datasets that contains text reviews for restaurants, information about users and information about businesses. After randomly choosing 50 users out of the dataset, our data contains 272 pieces of reviews of 251 restaurants and 50 users.

Table 1 shows descriptive information of our dataset.

Dataset	Size	Variables of Interest
Yelp _user	1M users *(50 as sample)*	**User_id**: "encrypted user id" **Name**: "first name"
Yelp_review	2.93GB *(272 texts as sample)*	**Review_id**: "encrypted review id" **User_id**: "encrypted user id" **Business_id**: "encrypted business id" **Text**: "review text"
Yelp_ business	144,070 Restaurants *(251 as sample)*	**Business_id**: "encrypted business id" **Name**: "business name" **Address:** "full address" **Stars**: star rating **City**: "city"

Table 1 Dataset Descriptive Summary

Procedure

Figure 1 shows our architectural procedure for this project. We note that the following steps were done separately on users and restaurants, except for the third step.

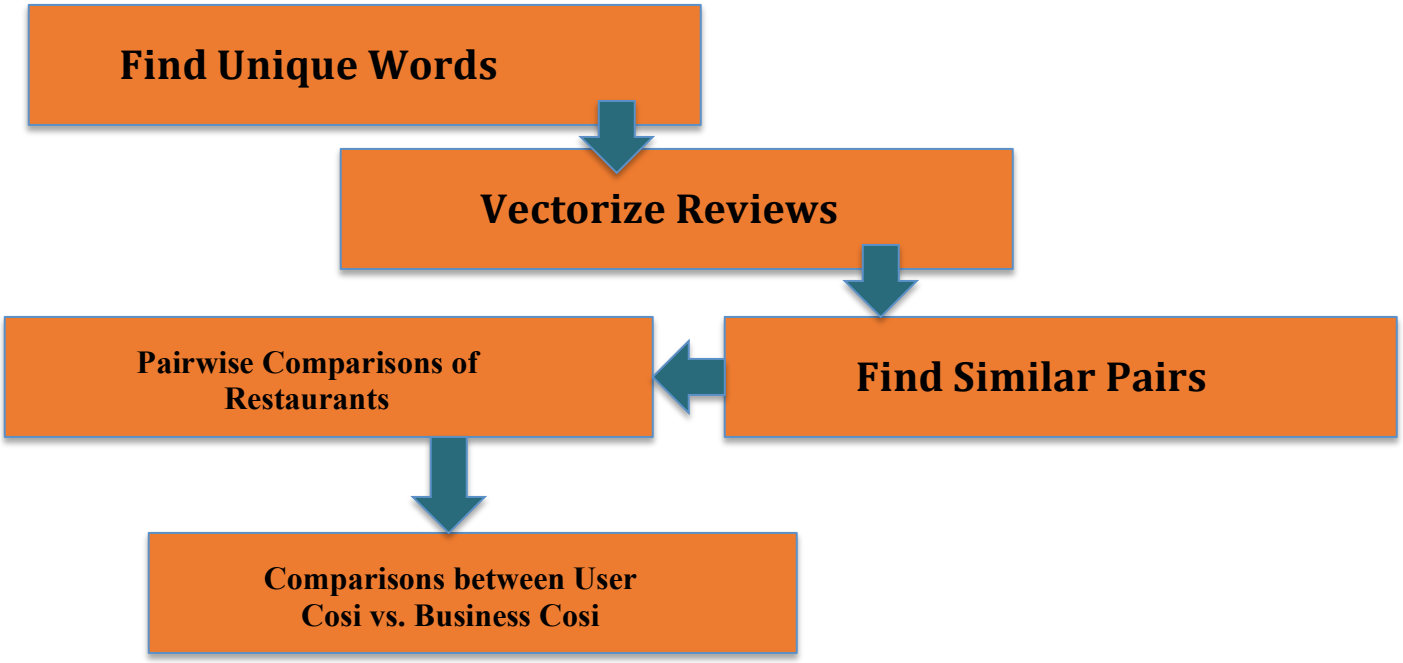

Figure 1 Architectural Procedure

Find Unique Words. In this task, we aggregated all of the 271 pieces of reviews and found the unique words, putting the words into a list; we call it the base vector. To perform this task, we used Map Reduce with None as its key and a list of unique words as its value.

Vectorize Reviews. For each user/restaurant, we aggregated all the reviews for this user/restaurant; map it with the base vector, and generate a vector for its own review words. We also use Map Reduce in this task, with user_id/business_id as the key and its vector (with the same length of the base vector) as the value. Whenever we went into a new word, we go to its slot in the base vector, and add 1 to its slot (frequency).

Find Similar Pair of Users. In this task we first paired each user with all the other users, and

then used "Cosi Similarity" method to calculate a Cosi similarity between each pairs. The end result is to find the one user that is most similar with our subject of interest, for each user. We also performed Map Reduce in this task. Input was a csv file with all possible pairs of user_ids. We connected Map Reduce job to a database, that contains the vector of each user, and then we would be able to calculate cosi similarities between all pairs of users. From that, for each user, we picked the other user that generated the largest cosi similarity.

Pairwise Comparisons of Restaurants. After we had our result of all most similar pairs of users, we went back to our database that included the visited restaurants for each user. For each pair of users in our resulting table, we pairwised every restaurant that they have been to, and calculated cosi similarities between these pairs of restaurants.

E.g. User 1 went to A, B, C. User 2 (which is the most similar user to User 1) went to B, C, D. For further comparison, we would pairwise all the restaurants: AB, AC, AD, BC, BD, CD

Comparisons between User Cosi vs. Business Cosi. For each pair of similar users, we compared each of he paired restaurants' cosi similarity with paired users' cosi similarity. Let's say the paired users' cosi similarity is β, then we would recommend the paired restaurants that have cosi similarities larger than (1- β). In other words, if β is small, meaning that the two users are not that similar, we would need to consider more, and pick restaurants that are very similar to them for better recommendation service.

Results

In this project, we set the significance level of similarity as $\alpha = 0.5$, which means that the

pairs that have a cosi similarity score higher than 0.5 would be classified as significant similar pair. The overall successful rate of the recommendation system is 0.16, meaning that we gave recommendations to 16% of the users; the successful rate for the significantly similar users is 0.16, and for the insignificantly similar users is 0.0.

Discussion

The results of this study could improve Yelp's recommendation system to a newer level. Right now their system is no more than physical aspect, such as location, cuisine or simple rating. However our aim is to generate a multidimensional recommendation, by our deep analysis for each piece of review.

This study also enriches our understanding of handling big dataset. First of all, we combined Map Reduce with database, to increase the efficiency of our work. We also utilized Google Cloud Computing, by offering access to large amounts of storage and the ability to rent the use of a cluster of machines for compute purposes, we were able to further decrease our runtime.

Limitation. The biggest challenge that we were facing was not being able to run some steps of this project on Google Dataproc to reduce our runtime. When we test 500 users out of 1M users, even if this amount is no more than 1% of the whole dataset, it still took us around 9 hours to generate the result. The complexity of this project, as well as the main point that confirms it as a big data project, is the pairwise comparison, which was also the part that took the longest time.

Reference

Hu, Minqing, Liu, Bing. 2004. Mining and Summarizing Customer Reviews. *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*

Pages 168-177. Seattle, WA, USA — August 22 - 25, 2004

C. Bird, (2015, May). Which local business reviews are better: Yelp or Google? [Online]

Available: http://localvox.com/blog/local-business-reviews-yelp-or-google/