Business Recommendations Based on Google Local Business Reviews

Timothy Lee, Benson Wu, Aiden Yoon; Table 22

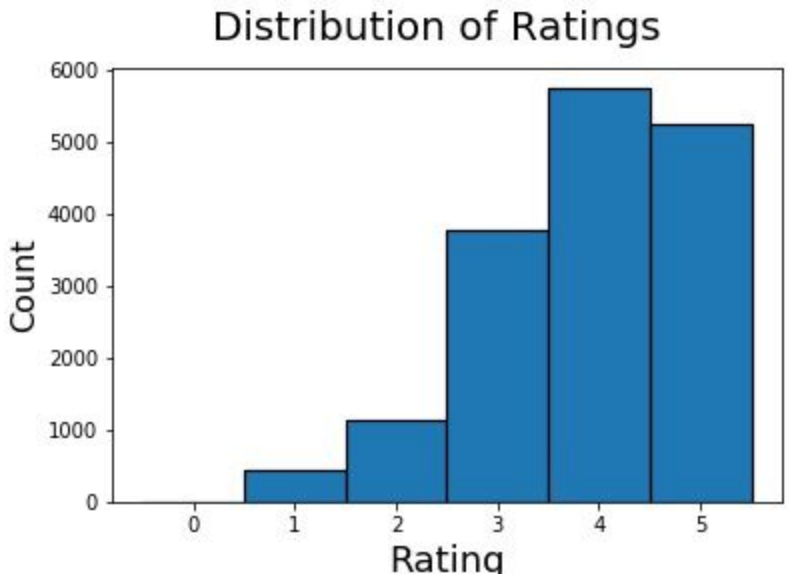
April 20, 2019

Introduction

In this analysis we attempt to develop a recommender system that can use business review data to recommend businesses to people based off of similar reviews/interests of other reviewers. Being able to learn from business review data could possibly allow people to more efficiently find products and content that they care about and would allow us to give the users a better experience in finding businesses they would potentially enjoy.

Data Description and Cleaning

Let's look at the distribution of ratings. Clearly, we see that most reviews tend to be on the higher end. However, this does not really help us, as we must consider the different types of businesses that people leave reviews for, the credibility of people who have only written a very minimal amount of reviews, and the left-skew in the distribution can be possibly explained by the tendency of reviewers to only leave reviews when they think something is excellent.



Due to the limitations of our machines, we were restricted to using a sample of the full data set. With this in mind, we proceed to filter our data to meet the following criteria: that each reviewer left *at least* 5 reviews. Our rationale behind this decision was to avoid cold start user profiles. It would be difficult to make accurate recommendations based off of these profiles.

Our goal was to create a user to user recommendation system. These were the steps that we took to make our data useable.

1) Create a user to business dataframe with the values in each row representing the rating that a user gave to the business they reviewed.
    a) In order to pivot our original data frame, we had to remove any instances where a user reviewed the same business twice.

gPlusPlaceId	1.0000085086337006e+20	1.0000119333884451e+20	1.000017139460348e+20	1.0000214164783343e+20	1.0000230763784749e+20
reviewerName					
A Google User	0.0	0.0	0.0	0.0	0.0
A Smith	0.0	0.0	0.0	0.0	0.0
AHsuan Chen	0.0	0.0	0.0	0.0	0.0
ALEXANDER PRIME	0.0	0.0	0.0	0.0	0.0
ALOK KUMAR	0.0	0.0	0.0	0.0	0.0
AMUSE THEMUSES	0.0	0.0	0.0	0.0	0.0
AUG ust	0.0	0.0	0.0	0.0	0.0
AYAKA OHKAWA (大川 綾香)	0.0	0.0	0.0	0.0	0.0
Aakash Garg	0.0	0.0	0.0	0.0	0.0
Aaron Babst	0.0	0.0	0.0	0.0	0.0
Aaron Berlin	0.0	0.0	0.0	0.0	0.0
Aaron Bevacqua	0.0	0.0	0.0	0.0	0.0

2) Next, we created a cosine similarity matrix. This matrix shows how similar two users are based on the ratings that they gave businesses. This is the formula that was used to calculate the values in the similarity matrix.

$$\text{similarity} = \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}}$$

Analysis

In order to find similar user groups, our next step was to develop a model that could find similarities between reviewers and then recommend businesses of interest based off of these similarities.

We used a K-Nearest Neighbors algorithm to find the *k* nearest neighbors based on the cosine similarity values. We decided to find the 3 nearest neighbors to prevent over saturation of recommendations

An example of our results is shown below:.

```
3 most similar users for User 120:

1: User 391, with similarity of 0.22958996631814121
2: User 984, with similarity of 0.015250371939424023
3: User 2648, with similarity of 0.006343127991661146
```

Although this model allows us to see similar users, we still do not have access to what businesses to recommend to the original user. Our next step was to look at each of the individual user-user matches and pull all businesses that the second user rated highly. With our example of User 391, this was their set of highly-rated (4 or 5 star) businesses:

	gPlusPlaceId	categories
4845	1.113850e+20	[Fish & Chips Restaurant, Fast Food Restaurant...
13482	1.150198e+20	[Sushi Restaurant, Japanese Restaurant, Takeou...
111994	1.129165e+20	[Vietnamese Restaurant, Asian Restaurant, Sout...
239870	1.119537e+20	[Steak House]
330633	1.142326e+20	[Cave, Tourist Attraction]

The problem with recommending all of these businesses to the original user is that some recommendations would not really make sense given the context of what the original user liked. For example, if the original user had no experience with Caves and Tourist attractions, recommending this business to them would not make sense. Therefore, our next step to provide accurate recommendations was to look into the categories of each business.

We took both the original user and the matched user's reviews and kept only the 4 and 5 star reviews. We proceeded to split the categories and converted our data into tall-data, where each row represented an individual category that a businesses was classified as.

	index	0
0	Gastropub	1.147667e+20
1	Wine Bar	1.147667e+20
2	American Restaurant	1.147667e+20
3	Pub	1.156924e+20
4	Restaurant	1.038538e+20
5	Bar	1.075091e+20
6	Italian Restaurant	1.094746e+20
7	European Restaurant	1.094746e+20
8	Bar	1.094746e+20
9	SCUBA Instructor	1.132213e+20
10	SCUBA Tour Agency	1.132213e+20
11	Vacation Home Rental Agency	1.132213e+20
12	Asian Restaurant	1.097801e+20
13	South Asian Restaurant	1.097801e+20
14	Indian Restaurant	1.097801e+20
15	Cafes and Snack Bars	1.053797e+20

	index	0
0	Fish & Chips Restaurant	1.113850e+20
1	Fast Food Restaurant	1.113850e+20
2	Fish and Chips Takeaway	1.113850e+20
3	Sushi Restaurant	1.150198e+20
4	Japanese Restaurant	1.150198e+20
5	Takeout Restaurant	1.150198e+20
6	Vietnamese Restaurant	1.129165e+20
7	Asian Restaurant	1.129165e+20
8	Southeast Asian Restaurant	1.129165e+20
9	Steak House	1.119537e+20
10	Cave	1.142326e+20
11	Tourist Attraction	1.142326e+20

Left Figure shows original user | **Right** figure shows matched user

The next step in our analysis for recommendation was to see if the businesses categories had some similarity between the two users. We proceeded by using cosine

analysis again; however, the goal this time was to use string similarity and compare the business categories against each other. Anything that was above a .25 threshold of similarity was marked as a similar establishment. Our threshold for .25 was used because two different categories of restaurants (i.e Asian and American) were given a cosine similarity value of about .263. Our goal was to see if similar establishments (in this case food industry) would be recommended to the other user, and so we decided to cap the threshold at .25.

Once all the categories were compared, we extracted all the gPlusPlaceID's where there existed a categorical match. These were are recommendations we generated for the original user:

categories	gPlusPlaceId
[Gastropub, Wine Bar, American Restaurant]	1.147667e+20
[Restaurant]	1.038538e+20
[Italian Restaurant, European Restaurant, Bar]	1.094746e+20
[Asian Restaurant, South Asian Restaurant, Ind...	1.097801e+20
[Gastropub, Wine Bar, American Restaurant]	1.147667e+20

Recommended Businesses along with Categories

Validation

Because we did not train the K Nearest Neighbors algorithm, there was no validation step. However, we can still judge the quality of the algorithm's prediction by examining the predictions against the correct labels of each observation.

Final Insights

We wanted to use the comment reviews to perform sentiment analysis and see which business proved to be the most satisfying for users. We believe that this would improve the recommender system as it would improve with a more detailed review of the business along with the number ratings. Another possible way that could help us better assess the quality of our recommender system is by gathering data on whether or not users actually sought after the recommended businesses, and whether they left a positive or negative review at these businesses. In doing so, this would allow us to incorporate that data into our K-Nearest Neighbors algorithm to give better recommendations not only on similarity to other reviewers, but also on whether or not the recommendations were accepted. This would inevitably give us the ability to truly test how efficient and effective our work is and see how we could make it better; we might be able to incorporate more statistical testing such as A,B testing to see if our model is good. Overall, our future goals for this project would be to develop a more accurate algorithm by using the entire data set. Because we were limited by our technology, we were unable to get the full potential for this project.