DS 501: Case Study 3

**Business Intelligence with Yelp.com**

Group 6:

Elizabeth Karpinski

Jingnan Xu

Qian Wang

Xiaoshuai Li

**Introduction**

Yelp is an international business based in California that crowd sources reviews about local businesses. Users can write reviews for a business and rank it from one to five stars, so that potential customers considering frequenting the business can learn from their experience. Yelp basic search functionality allows users to search both by type of business and by location. Yelp advanced search functionality allows users to narrow down by more filtering options; in the case of restaurants, these may include filtering by places that do delivery or take reservations, places that are currently open, the expensiveness of the place, and even user ‘tags’ about the kind of food served, such as ‘Italian’ or ‘brunch.’ This helps to give users an ability to look for the kind of businesses they are looking for, and also to gauge the business’s performance. This is the main service Yelp provides to its users on the consumer side. Small businesses also benefit from Yelp because it can help to promote their business, raising both awareness and its reputation.

It is a logical extension of this to consider that Yelp’s ability to recommend businesses as well as to understand what makes a business likely to be ranked highly by a certain user could be an enormous advantage to its business model. Additionally, it is common knowledge that many other popular internet businesses, such as Amazon and Netflix, thrive by use of their recommendation systems. In fact, for five years, Netflix had a $1 million dollar prize on whoever could improve their recommendation system by just 10%. This, more than anything else, shows how lucrative recommendation systems can be both on the consumer and on the commercial side of things. Applying a similar recommendation system to Yelp and their data set is the question we chose to tackle for our report.

**Data Collection and Methodology**

Like many prosperous internet companies, Yelp amasses data at an amazing rate. As part of a data hackathon challenge, Yelp makes part of this data set available to students who are interested in analyzing it and attempting to answer a business intelligence question as part of a competition. This data is provided in .JSON format, and includes information on hundreds of thousands of users, users’ reviews, and businesses.

In order to begin analysis on the data to answer our question, we chose the collaborative filtering method of generating recommendations out of three popular recommender systems. Collaborative filtering has the advantage of learning market segments, though it has problems with cold starts (cannot be used with insufficient data). Since Yelp already has a data set, cold starts would not be a problem. We also considered content-based systems, which have the advantage of not requiring community and can compare between items, but require content descriptions and also suffer with cold starts. The final type of system we considered was knowledge-based, which creates deterministic recommendations and assured quality, but does not react to short term trends. After carefully weighing the pros and cons of the three types of recommender systems, we deemed the collaborative filtering system the best fit for the Yelp data set.

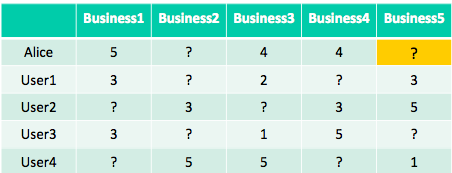
We used Spark Core API and SQL to handle the data. Apache Spark is a fast, generalized engine for large-scale data processing. We chose it for its ease of use, speed, and unified engine capabilities. It is considered an up and coming competitor to Hadoop. which for some processes like logistic regression, it can run up to a hundred times faster than.



Figure

**Statistical Modeling**

Collaborative filtering is a way of generating recommendations with large data sets that are typically user centered. It analyzes user rankings and reviews for similarities in their past rankings and reviews and uses those similarities to predict rankings and reviews that a user hasn’t yet done. Since the reviews and the rankings represent separate data sets, we created two separate collaborative filtering recommendation models: one focusing on the review based data and one on the ranking based data.



Figure

For the first part, we focused on user ranking data (see figure 2 for an example matrix). We used the memory-based variant of this technique, which focuses on the user rating data to compute similarity. In this method, the Pearson correlation, see below, equation gives us the similarity of two users.

The similarity is used to choose a number of nearest neighbors, who are then used to make predictions about the users’ preferences, using the following equation.

Within these equations, we used the following for the variables

*i = the ith business*

*Va,i = user a’s rank of the ith business*

*Va = the mean rank of user a*

*Vu,i = user u’s rank of the ith business*

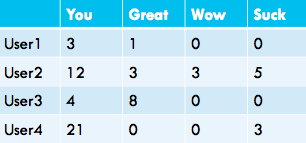
*Yu = the mean rank of user u*

*σa= the standard deviation of user a*

*σu = the standard deviation of user u*

*pa,I = prediction score*

**Implementing the collaborative filtering method based on review similarity was done using a similar process, but with a focus on lexical similarity between users. The variables represented distinct words in the reviews rather than rankings from individual businesses (see figure 3 for an example table of this data) and ran the same process on that.**



Figure

**A final part of this process was neighbor selection. For a given active user, we needed to select a number of correlated users to serve as a source of predictions: the number of nearest neighbors. There are two common approaches, both of which we experimented with. First, the standard approach, to choose a set number n users’ ranks based on similarity weights; second, to include all users whose similarity weight is above a given threshold.**

**Challenges**

However, there were a number of challenges we faced in implementing the collaborative filtering method. First, although the dataset from Yelp included 1.5 million reviews, 61 thousand businesses, and 366 thousand users, the important data was surprisingly sparse. Using those numbers, the average reviewer only has 5 reviews; however, the data is skewed highly to the right, since some Yelp reviewers are prolific and others are almost entirely inactive. This was a problem for us because data sparsely is a common failure point of the collaborative filtering method. First, because the system relies on users past preferences, users need to have sufficient past preferences on the record for us to be able to look for similarities and to extrapolate; without enough reviews, it is difficult to find sufficient similarities between users or the reviews they have submitted. This cascades into also making it difficult to predict, since the overabundance of negative signals and dissimilarity can confuse the algorithm.

Another problem we considered in our implementation was the frequency of fake reviews which might hurt the accuracy and integrity of our algorithm’s results. Fake reviews is another problem that the Yelp business model has struggled with, as small businesses have a great incentive to artificially inflate their own reputation and sabotage their competitors by issuing falsified reviews online. The problem of detecting these fake reviews is made additionally complicated by the myriad of methods used in order to create fake reviews. For example, people may use spam bots, hire a ‘reputation management’ company, or even just order employees to fill the company yelp page with glowing reviews. Since each of these approaches appears very different in practice and exhibits different signs, this means that Yelp often struggles to detect and remove fake reviews on small businesses. However, one trait that all methods of generating fake reviews tend to have in common is that the accounts on which they are made do not have a large number of reviews. Most will only have one or two, on the target company. In order to account for this and to mitigate its effect, we limited the sample of users to use in our collaborative filtering algorithm to prolific Yelp users with 80 or more reviews. This also helped to resolve the problem of sparse data, although the data was still not at the ideal level of abundance for the algorithm.

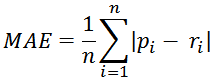
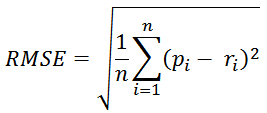
|  |  |
| --- | --- |
|  |  |
| All Users | Users Who Rank more than 80 Businesses |

**Example Implementation**

**For an example, we implemented a rank based collaborative filtering model with the top 45 neighbors. As seen in the discussion section, this was the strategy with the lowest root mean square error. For this example, we chose a specific user with ID “‘fczQCSmaWF78toLEmb0Zsw’ and the username “Gabi” and predicted their rating for a specific business, “Rudy’s Country Store and Bar-B-Q”, which is located in Chandler, Arizona. Rudy’s Country Store has an average of four stars and 391 reviews. According to our calculations, however, since Gabi is more similar to the users to have rated it highly, Gabi is likely to give it a full five stars.**

**Results and Discussion**

We used two different accuracy measures to examine the error rate between our two different collaborative filtering recommendation models. The mean absolute error computes the deviation between predicted ratings and actual ratings. The root mean square error is similar to the mean absolute error, but it places more emphasis on larger deviation. This is the measure we chose to focus on in a quantitative evaluation of our two models.

Figure

Charts of the root mean square error can be seen below. In figures 5 and 6, the root mean square error is represented along the Y axis, while the number of neighbors used in the algorithm is plotted along the X axis. As can be seen in the figures, the two different models had a different ideal number of nearest neighbors. In the review based RMSE, the model with 20 nearest neighbors had the lowest RMSE. In the rank based RMSE, the model with 45 different nearest neighbors had the lowest RMSE. Additionally, while the two models were relatively close, the rank based model outdid the review based model with an RMSE of .76 vs. an RMSE of .80.



Figure 5 (Review based rmse)



Figure 6 (rank based rmse)

A less quantitative concern about the integrity of our results stems from the similarity that the collaborative filtering detects appears to be location-based; that is, the best indicator of whether a user is likely to give a high ranking to a business is if the user in in the same area as the business. While users may occasionally review businesses outside of their local area (e.g., on vacation), almost all of their reviews will still be centered within their area. This means that a portion of their similarity to other local users will be from their shared location, rather than shared taste.