With A Little Help From Yelp

Caitlin Bonnar
Computer Science
University of Washington
cbonnar@cs.uw.edu

Felicia Cordeiro
Computer Science
University of Washington
Felicia0@cs.uw.edu

Julie Michelman
Statistics
University of Washington
michelmj@uw.edu

INTRODUCTION

Today, people commonly use applications that allow them to search for the best restaurants (Yelp), hotels (TripAdvisor), plane tickets (Expedia), and merchandise (Amazon). These applications, although extremely useful for narrowing down a large list of data to a few selections that meet certain user-specified criteria, are not very good at providing a general overview or aggregation of the data. For example, Yelp provides options such as neighborhood, distance from a certain location, features (like whether they have a bar, good for kids, or have free wifi), price, and category (e.g., restaurant, fast food, seafood, etc.). The user can then select the options that they would like and in return receive a list of restaurants that meet these requirements. Although this method shrinks a large number of possibilities to a small group of options that the user can select from (see Figure 1), the user is not given any form of overview except for a map with the locations of the restaurants that the query returned.

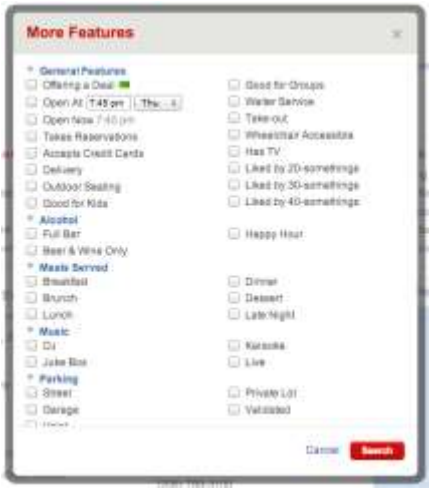

Figure 1: A screenshot from Yelp where users can narrow down their results by different features

The Yelp dataset consists of information on businesses, reviews, and users. Business attributes include location, categories, average rating, and review count. Review attributes include rating, text of the review, and votes for useful, funny, and cool. User attributes include review count, average rating, and votes received. We run queries on this information to provide the user with a visual overview of their search results as well as an overview of the neighborhoods they are searching in, which will aid the user in making a well-informed decision.

We combine current Yelp data from greater Phoenix, Arizona area with information about the neighborhoods of Phoenix to create an interactive map that provides an overview of Phoenix businesses, citizens, and neighborhoods. The data are split among three primary categories: Businesses, Users leaving reviews on businesses, and Reviews left on businesses. Each query that we ran on the website returned results for the general Phoenix area (the 'Overview' page), for each particular neighborhood on our interactive map, and for each category of business (both by neighborhood and overall) The following are examples of queries that we run in each category:

Businesses

- What businesses have the highest rating as a function of average star rating and total review count?
- Which businesses are closed? Which neighborhoods have the highest percentage of closed businesses?
- What are the most popular neighborhoods for bars? Coffee shops?

Users

- Who are the funniest reviewers? Most useful reviewers? 'Coolest' reviewers?
- Which users have written the most reviews in the greater Phoenix area?

Reviews

- What are the top funny reviews in each area/category? Useful? Cool?
- Which categories have the greatest total count of reviews? Highest average?

RELATED WORK

This project encompasses a variety of topics in data mining, HCI, and data visualization. Data mining techniques have been used to mine and summarize opinions from customer reviews [5], as well as determine fake reviews on TripAdvisor and Expedia [3,4]. Our work differs in that we summarize user, business, and neighborhood information and attempt to discover relationships between them. To provide our visual overview, we draw on techniques discussed in data visualization and user interface integration within the field of human-computer interaction [2].

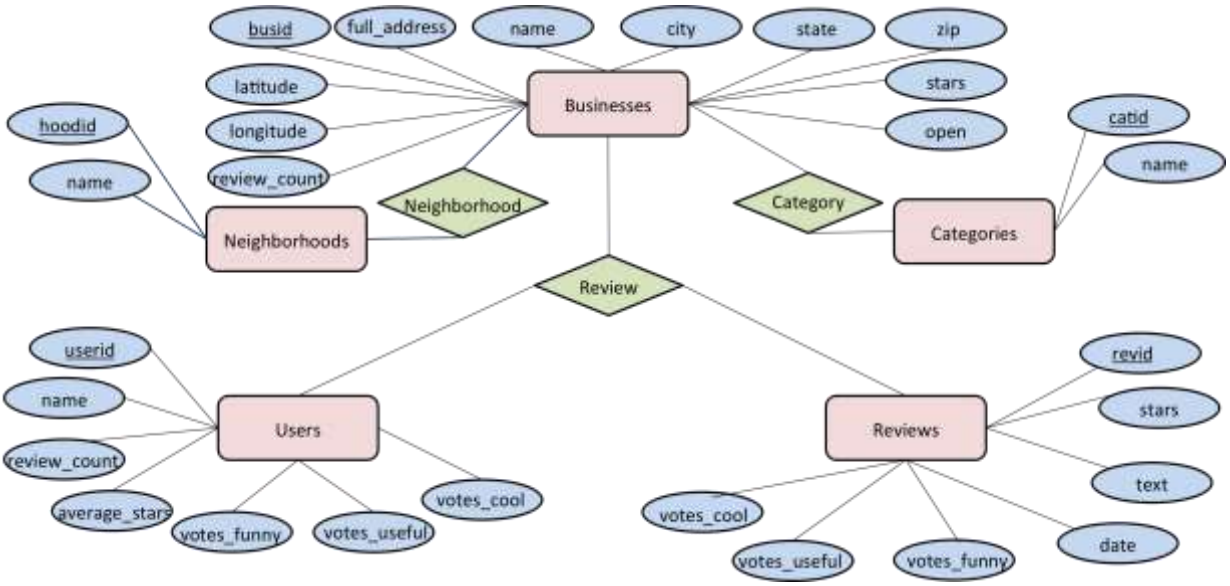

Figure 2: ER Diagram for Yelp Schema

OUR APPROACH

Our project was completed in several parts. First, we created a well-designed schema and corresponding E/R diagram (see Figure 2). We'd like to porint out the three-way relationship 'Review.' This relationship allows a user to review the same business multiple times. Although we only found this to be the case once in our sample of Yelp data, we thought it was an important feature to support. Next we need to import the Yelp data into Postgres following the structure of the E/R diagram.

The first step to importing our data into Postgres was to create a Python wrapper to that converted the Yelp data formatted as JSON to text files that could be imported into the following tables: Businesses, Users, Review, Reviews, ReviewText, Category, Categories, Neighborhood, and Neighborhoods; where the Neighborhood, Category, and Review tables represent the relationships between the tables. We experienced some difficulties when converting the Yelp JSON files into text files.

1. Some of the Yelp data items contained arrays or lists
2. Some items were simply missing (which threw an error).
3. The reviews contained tabs, line breaks, and special characters that when imported into Postgres also gave errors.
4. We first attempted to convert the JSON to XML, however this required a lot of restructuring and thus went directly to text.

The second step to importing the data to Postgres was to create initial (temp) tables for businesses, users, and reviews that contained rows of key-value pairs, (ID, attributeName, value). Since Yelp does not provide an id for each review, we created a sequence that incremented for each row. Next, we transformed our temporary tables to match our well-defined schema. In the process, we discovered that none of the businesses had neighborhoods listed, and planned to use zip code as a surrogate. Thus we parsed the full address in the Python file to extract the zip code for each business. For businesses without zip codes, we wrote an empty string to the text file and converted this to a NULL in the Businesses table. We also discovered that some of the reviews had Yelp user id's that are not among the set of users in the data. We added these to the Users table with all other fields NULL. This allows user id to be a foreign key in the Review table. This also allows us to distinguish among different 'null' users, which is important in summary statistics. For example, one of the 'null' users is among the top reviewers and we know all these reviews were written by the same person, even though we only know their user id.

Our project idea was largely based on using the neighborhood data that Yelp promised us, and although we had the idea to substitute it with zip code, when it came down to implementation, this was not a good idea since multiple neighborhoods share multiple zip codes. To solve the neighborhood dilemma, we searched for other methods to acquire the data. Our first attempt was to send http requests to Google for each business' full address with the hope that it would return the neighborhood data; however, this was not the case and they also had daily limits. Our second idea was to send an http request to Mapfluence, a Javascript API that returns the neighborhood of a location from longitude and latitude. However, we requested an API key several times with no response. Finally, we found out

```
SELECT b.city, n.name
FROM neighborhoods n, businesses b
WHERE ST_CONTAINS(n.the_geom, GeomFromText('POINT(' || b.latitude||' '|| b.longitude ||')', -1));
```

Figure 3: Query to Determine the Neighborhood of a Business Based on Lat/Long

that Zillow has compiled shp files for the neighborhoods of various states throughout the country, including Arizona. We then installed PostGIS, converted the shp file to a SQL file and then created a neighborhood table that contained the boundaries for each neighborhood of Arizona. Finally, we joined the neighborhood PostGIS table with ours, and we queried the longitude and latitude of every business to determine which neighborhood boundary contained it.

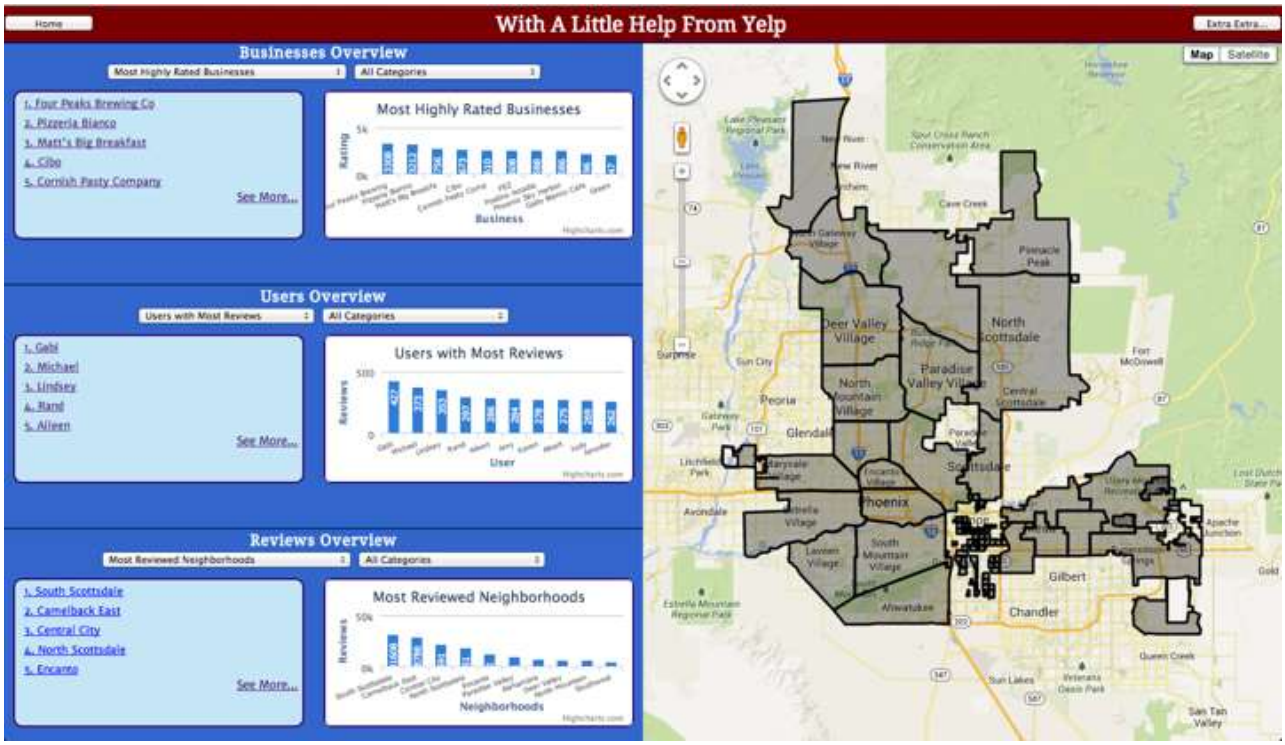

Figure 4: WithALittleHelpFromYelp Interface

Our Implementation

We built a website using Node.js, Javascript, Ajax, JQuery, HTML, and CSS that is hosted on a Linode server. The site can be accessed at www.WithALittleHelpFromYelp.com (see Figure 4). We used a Node.js framework, Node-Postgres, to connect to our local Postgres database and run queries. We used the Google Maps Javascript API to include a map, and added a GeoJSON overlay that we converted from PostGIS shape files. In order to link the GeoJSON coordinates to the correct Google Map coordinates, we used a library called GeoJSON.js that allowed us to import our GeoJSON file and create an object that could be overlaid correctly on the Google map.

Features

We implemented several features into our site to provide the user of an overview of the Yelp data.

1) 12 options for the types of data [(Businesses, Users, Reviews) X (Overview, Neighborhood) X (All Categories, Specific Categories)]

2) An interactive map displaying the different neighborhoods in Phoenix and surrounding area. Users can click on specific neighborhoods in order to narrow queries for businesses and users in that neighborhood

3) Bar graphs that accompany the top results and provide more (quantified) information about the selected queries (shown to the left of Figure 4).

4) Links to the corresponding Yelp page for the businesses listed, an expanded view for query results, and a list of the businesses reviewed by the users listed (see Figure 6)

Figure 5: Clicking on a business query result allows the user to link back to the detailed page on Yelp

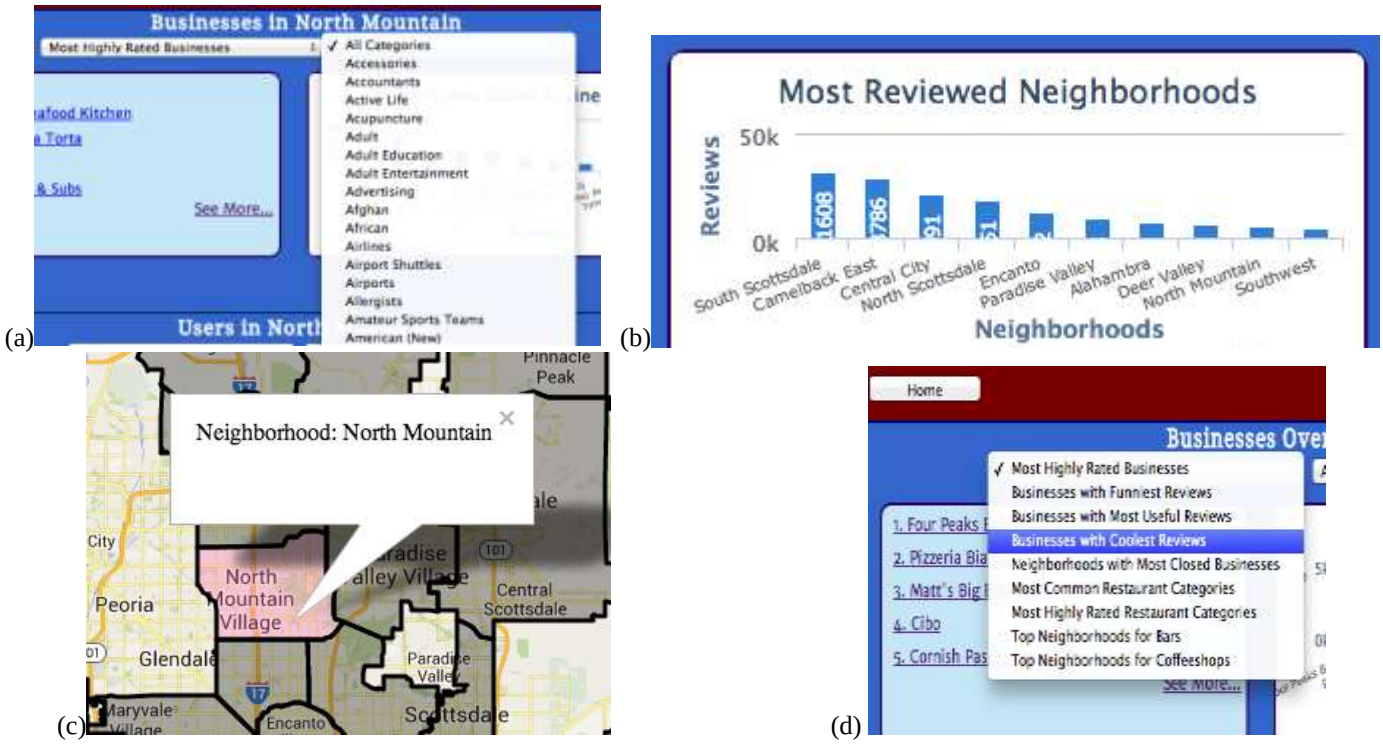

Figure 6: Showing a selection of the many different queries and categories available

5) Additional statistical analysis that can be reached by clicking on the "Extra, extra" button (Discussed in following section)

Statistical Analysis of Yelp Data

We use statistical analysis to address three questions of interest about the Yelp data:

1) Do more critical users write more useful reviews?
2) Can we use Yelp data to predict if a business will close?
3) Do reviewers stick to the same type of business?

We constructed a Restaurant indicator variable to use in these analyses. We hand-labeled each category as a type of restaurant/bar or not. For each business, if at least one of its categories is a restaurant category, we say that business is a restaurant.

Critical and Useful

The first question asks if more critical users write more useful reviews. In particular, we want to know whether users with a lower average star rating receive more useful votes on their reviews. A user's useful vote count is strongly correlated with their review count. Therefore, we adjust for review count in the model.

Review count and useful vote count are both highly skewed to the right. To address this, we omit users with 0 for either count and apply log transformations. Then we fit a linear regression model.

$$\log(UsefulVotes) = \beta_0 + \beta_1 \log(ReviewCount) + \beta_2 AverageStarRating + \epsilon$$

The coefficient estimates are presented in Figure 7, along with their standard errors and p-values.

Parameter	Estimate	Std. Error	p-value
β_0	0.324	0.015	$< 10^{-12}$
β_1	0.868	0.004	$< 10^{-12}$
β_2	-0.048	0.004	$< 10^{-12}$

Figure 7: Coefficient estimates for useful votes model

The coefficient of *StarRating* is significantly less than zero. This is evidence that on average more critical reviewers receive more useful votes. However, the coefficient is very small in absolute value. Suppose two reviewers have the same review count but one has twice the average star rating (a large difference). That reviewer is expected to have $2^{-0.048}$ =0.967 times as many useful votes as the other. As illustrated in Figure 8, the effect of average star rating on number of useful votes is negligible in practice. The discrepancy between the practical and statistical conclusions can be attributed to the very large number of data points (~38,000 users).

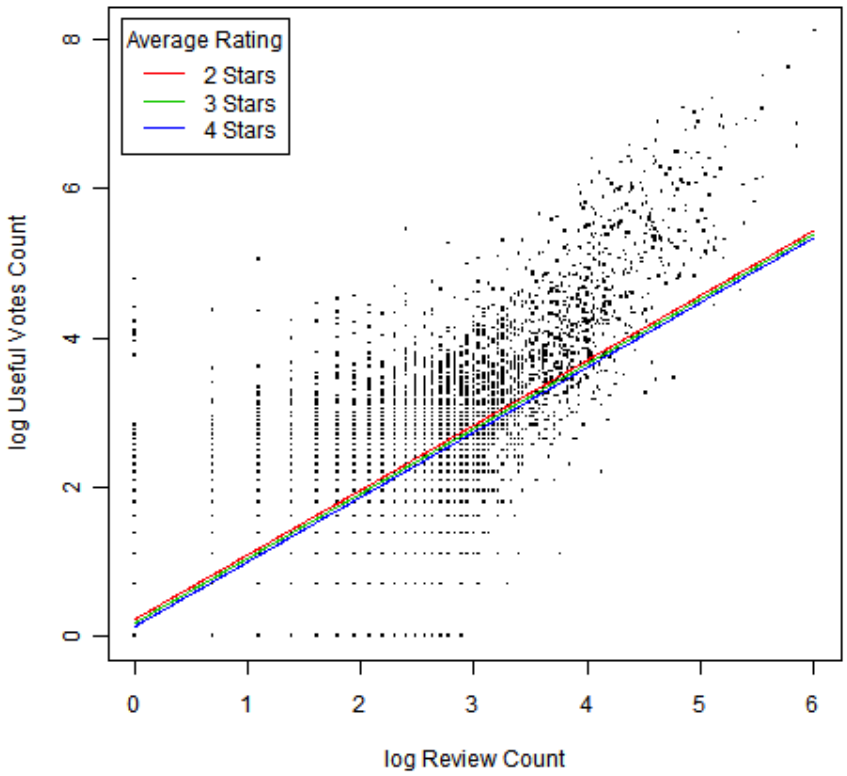

Figure 8: Scatterplot of useful vote count versus review count with fitted lines for several average ratings

Closed Businesses

The second question asks if we can use the Yelp data to predict whether a business will close. In particular, we would like to see how some of the features available in our data are associated with whether or not a business is open. To do this, we fit a logistic regression for the binary response open (1) or closed (0). The features under consideration are the review count and star rating for the business and an indicator of whether it is a restaurant (or bar). Logistic regression works best when approximately half of the data points have response 1 and half 0. About 90% of the businesses in the Yelp data are open. To create a balanced data set for the model, we use all the closed businesses and equal sized sample of the open businesses. The logistic regression model is

$$logit(\Pr(Open)) = \beta_0 + \beta_1 \log(ReviewCount) + \beta_2 StarRating + \beta_3 Restaurant$$

The coefficient estimates presented in Figure 9, along with their standard errors and p-values. These coefficients suggest that businesses with more reviews or higher ratings are more likely to be open. Restaurants and bars are less likely to be open than other types of businesses. These trends are illustrated in the Figure 10(a), a plot of probability open versus log review count. The curves show the predicted probabilities for various combinations of rating and business type.

Parameter	Estimate	Std. Error	p-value
β_0	-0.248	0.228	0.276
β_1	0.370	0.044	$< 10^{-12}$
β_2	0.232	0.057	0.00005
β_3	-2.046	0.105	$< 10^{-12}$

Figure 9: Coefficient estimates for open/closed model

Review Specialization

The third question asks whether users tend to review many of the same type of business. For this question we will only look at whether reviews are for a restaurant or not. Of ~184,000 reviews in our data, 76.4% of them are for restaurants. Our null hypothesis is that users do not specialize in restaurants or non-restaurants. Under the null, the number of restaurant reviews for any user is a Binomial random variable with p=0.764 and n, their review count.

Figure 10(b) shows the number of reviews versus the number of reviews for restaurants. It also shows the expected restaurant review count and the 95% interval bounds under the null. We only look at users with at least 10 reviews because the interval does not make much sense for smaller review counts. Among these users, only about 73.8% had their number of restaurant reviews fall in the bounds for their respective review count. This is much

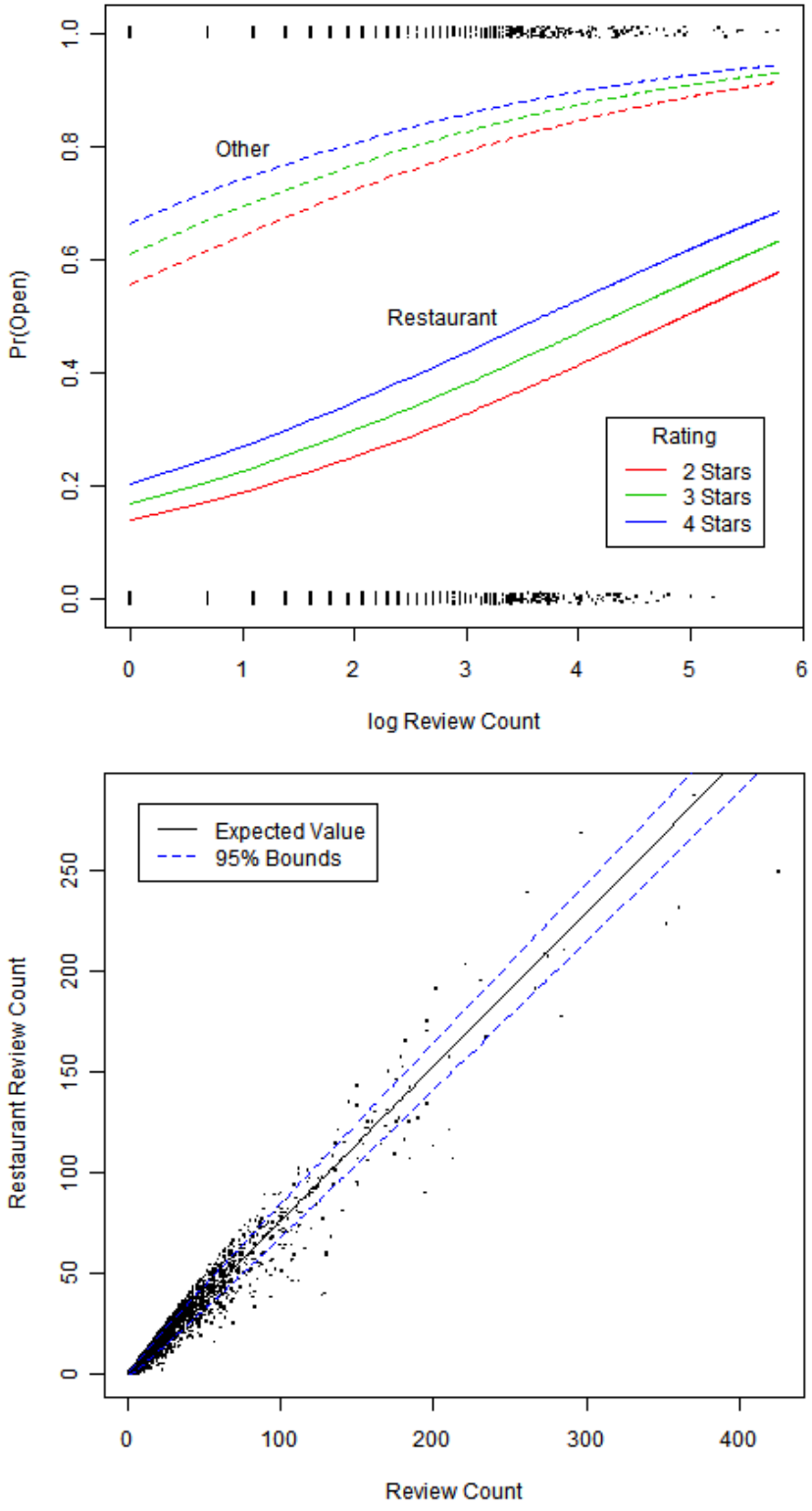

Figure 10: (a) Scatterplot of open status versus review count with fitted curves for several rating/ business type combinations. (b) Scatterplot of restaurant review count versus review count with mean and 95% bounds under the null hypothesis of no specialization in reviewing restaurants or non-restaurants.

lower than the 95% we would expect if there were no specialization. We reject the null hypothesis. We conclude that the rate of reviewing restaurants versus other types of businesses varies among users.

CONCLUSION

We present 'With A Little Help From Yelp,' an interface that allows users to quickly view aggregated Yelp data in order to discover interesting trends within different neighborhoods, types of businesses, and the greater Phoenix area. We combined the Zillow dataset and existing GeoJSON shp files for Arizona with our data in order to run queries in multiple dimensions. However, the queries we displayed on the site, while allowing for a large scope of information, were generally simple; thus, we came up with additional queries that allowed us to look for interesting correlations. Overall, we believe this work can be extended to provide even more helpful and interesting information to users based on the Yelp dataset.

REFERENCES

[1] Che, A, Hernandez, G, Hintze, M, McAdam, R. Stufte: http://lab.thegilby.com/projects/stufte/

[2] Daniel, Florian, Yu, Jin, Benatallah, Boualem, Casati, Fabio, Matera, Maristella, and Saint-Paul, Regis. 2007. Understanding UI Integration: A Survey of Problems, Technologies, and Opportunities. IEEE Internet Computing 11, 3 (May 2007).

[3] Hu, Nan, et al. "Manipulation of online reviews: An analysis of ratings, readability, and sentiments." Decision Support Systems 52.3 (2012): 674-684.

[4] Mayzlin, Dina, Yaniv Dover, and Judith A. Chevalier. Promotional reviews: An empirical investigation of online review manipulation. No. w18340. National Bureau of Economic Research, 2012.

[5] Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining (KDD '04). ACM, New York, NY, USA, 168-177.

[6] https://github.com/brianc/node-postgres

[7] http://www.yelp.com/dataset_challenge/