Lab 3 – Question 8 – Info 290-T Data Mining & Analytics

Michael Ball , Alec Guertin, Peter Sujan

Dataset:

We are using a dataset of approximately 220,000 Yelp reviews. These reviews are stored in the file `yelp_reviews.txt` on the class server. Each item in this file contains information relevant to a unique user review posted on the Yelp website. The reviews include data such as business IDs, user IDs, review IDs, the text of the review, and other general information. For our purposes, we are primarily interested in just a few of these features, namely the date of the review, the stars rating, the useful votes, the cool votes and the funny votes given to each review. Before analyzing our data we grouped all reviews by users' unique IDs to generate a plethora of feature vectors based on metrics for each user, like average length of review, or total number of reviews. These features are contained in the file yelp_reviewers.txt. These features include data such as the total ratings and votes for each reviewer, the natural log of these values, the percentage of each type of vote to the total votes (per user) and the users' most active years.

Research Question:

The fundamental question our group aims to answer is:

> *Do a user's reviews on Yelp improve as a user submits more reviews? Are these reviews better received by other users on Yelp?* ***Essentially****: Does a reviewer get better with practice? Can we use these features to determine the "quality" of a reviewer on Yelp? Furthermore, are there other qualitative differences between "good" and "bad" reviewers, such as being verbose, funny, or cool?*

Features Selected:

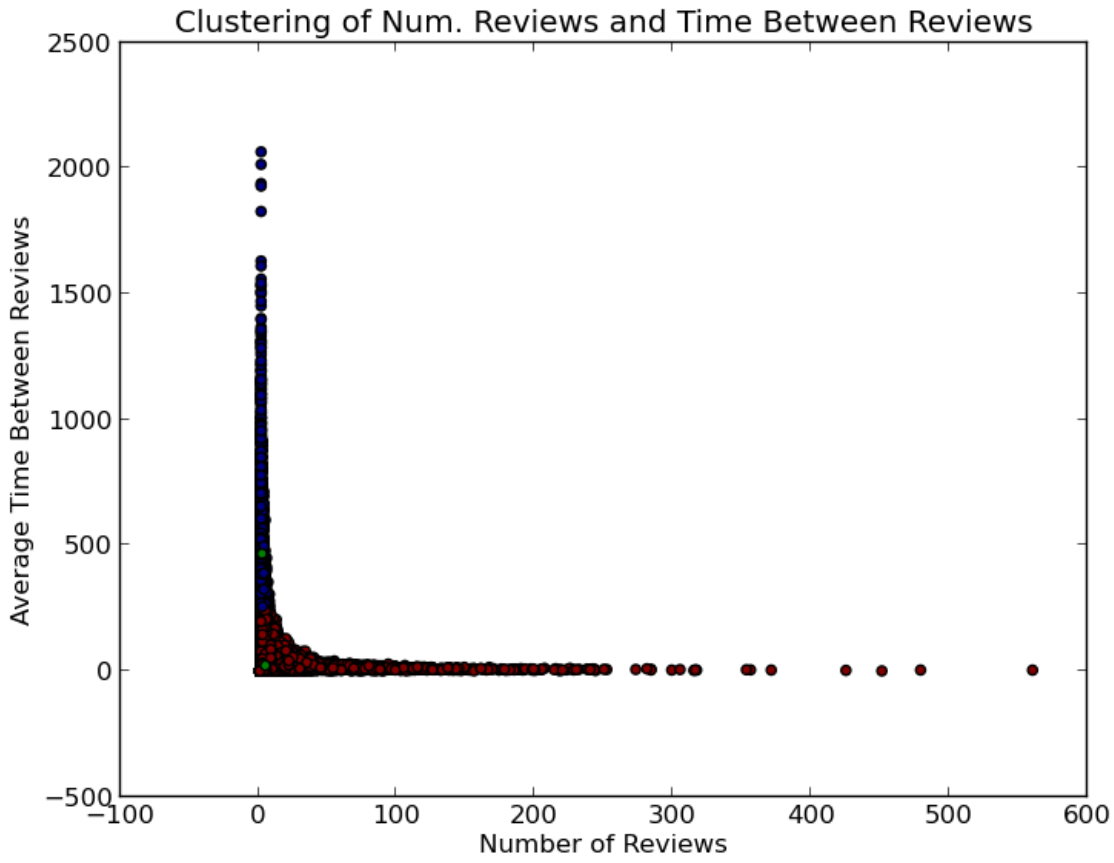
- Length of Review: The feature is the average length of, in characters, of all reviews submitted by a reviewer.
- Frequency of Review: The frequency of reviews is really two metrics. We use both the total number of reviews by a reviewer, as well as the average time between a reviewers reviews.
- Average Star Rating: This feature is the average star rating a reviewer awarded over all businesses they reviewed.
- Votes {Usel, Cool, Funny} Awarded: On Yelp, anyone can vote a review as any combination of "Useful", "Funny", or "Cool". These features describe the average votes awarded to all reviews submitted by a reviewer.

Methods:

To explore the questions posed above, we decided to cluster the data using the k-means algorithm. We began by clustering on an assortment of features including the number of reviews, useful votes per review, average review character length, days active on Yelp, average number of days between reviews and number of unique words used by the reviewer. The motivation behind this choice of features was that they measure the "quality" of a reviewer. Our hope was that the clustering algorithm would detect groups of various quality. We ran k-means for $k = 2, \ldots, 8$ and found that two clusters had the highest silhouette coefficient. When comparing the centroid (mean) vectors of the two clusters, we noticed that many of the values were remarkably close to one-another. Since these features did not appear to help separate the clusters, we decided to re-run k-means using only the discriminating features (number of reviews and average time between reviews). The resulting mean vectors were: [5.00, 22.05] and [2.63, 466.74], indicating a "frequent reviewer" cluster and a "rare reviewer" cluster. The results of this clustering are displayed in the plot below. Cluster means are displayed in green. Using this clustering, we compared the cluster averages for various features. A table summarizing several of these features is shown below.

Cluster	Star Rating	Votes Useful[1]	Votes Cool[1]	Votes Funny[1]
Frequent	3.773078	0.358706	0.285208	0.812338
Infrequent	3.752425	0.374323	0.294615	0.877750

[1] These calculations are per-review.



Results:

Unfortunately, our analysis did not lead to any particularly insightful results. Nonetheless, we found this exploration interesting. First, the clustering did not uncover much structure underlying "reviewer quality," but simply divided users by how often they review. Second, comparing other features between the two clusters did not reveal substantial differences between the two clusters. In fact, all average feature values of the two clusters were almost identical. This is, however, not entirely surprising; using only number of reviews and time between reviews only yields rather broad, inhomogeneous clusters. As a consequence of our relatively poor clusters, we didn't see much difference in other features between the two clusters. Perhaps using a supervised learning method would facilitate learning of features that are more directly associated with reviewer quality, and give more meaningful distinctions between the different cluster.