CS224W Final Project: Predicting Yelp Ratings From Social Network Data

Do (kelvindo@), Rotimi Opeke (ropeke@), James Webb (jmwebb@)
Department of Computer Science, Stanford University

December 9, 2015

Abstract

Yelp is a popular consumer application that has consistently kept itself relevant by integrating business reviews within an extensive social network. While the ability to add friends and internally message other users isn't as strongly emphasized across the product, the social component of the application helps provide incentive for users contribute to the greater Yelp community. This network community, of both users and businesses creates an invaluable network of information from which future user usage can be modeled. This paper examines the relational Yelp data to make predictions about the actions that users would take on the network - specifically to make predictions on their future review scores. Through a study of the communities that exist within the graph and through several different similarity metrics we hope to serve unique predictions about a user's future review scores based on information made available through network analysis.

1 Introduction

Yelp is one of the more unique social networks in the sense that it thrives off of highly qualitative user-inputted reviews. Each one of these reviews, tailored to a multitude of businesses serve as an unbiased platform in which the demographic has an equal voice in ranking and comparing local businesses. With information about how certain users chose to review businesses and with information about what users are "friends" with others, the ability to make predictions about future reviews greatly increases.

To do this, we explore several similarity classifiers to determine how related users are in our social graph. Within our study, we leverage Jaccard Similarity, Clique Community Similarity, and Random Walk PageRank Similarity, and Centrality Vector Distance Similarity to calculate similarity between users.

Finally, we feed graph data into recommender systems to provide a good gauge of future outcomes. We implement a K Nearest Neighbors (kNN) regressor to predict ratings based on ratings given by similar users.

2 Prior Work

2.1 Effects of User Similarity in Social Media (Anderson, Huttenlocher, Kleinberg, Leskovec, 2012)

This paper explores the possibilities of predicting evaluations between users based on information about the users themselves, not their previous evaluation history. Specifically, for sites that allow user contribution to be rated, such as Wikipedia and StackOverflow, the research examined how a contribution would be received by a given audience. The researchers used a similarity score, based on overlapping interests between users, and studied how this corresponded with positive evaluation of user contributions.

2.2 Evaluating Collaborative Filtering Recommender Systems (Herlocker, Konstan, Terveen & Riedl, 2004)

In, Herlocker et. al. discuss the main considerations when evaluating recommender systems since many implementations consider a wide range of metrics and data sets. Recommendation goals can vary widely based on its service to its users. This paper highlights how its important to consider that some recommender system success can vary widely if you consider what the goal of the recommendation is (ex. looking for a movie to watch versus looking to browse through a news feed). Additionally there should be consideration that a data set is not too sparse and is appropriate to fit the goals of the recommendation.

2.3 Random-Walk Computation of Similarities between Nodes of a Graph (Fouss, Pirotte, Renders, Saerens, 2007)

This paper explores the possibility of relating nodes in a movie/viewer database through similarity measures. With this calculated similarity between viewers, between movies, and between viewers and movies as the engine behind a collaborator recommendation system. In particular, the research explored the use of random walks and Markov chains as a measure of similarity between two given nodes. This problem space relates to the random walks algorithms such as PageRank that have been discussed in class.

3 Data Set

3.1 Yelp Dataset Challenge

The original motivation was to use data from the Yelp Dataset Challenge to predict the reviews of the many users included in this dataset. The Yelp Dataset advertises 1.6M reviews by 366K users for 61K businesses worldwide. In the user friendship graph, there are over 2.9M social edges between the 366K users making for an average of 7.9 edges per user.

The following is the degree distribution graph for the entire Yelp Dataset Challenge dataset:

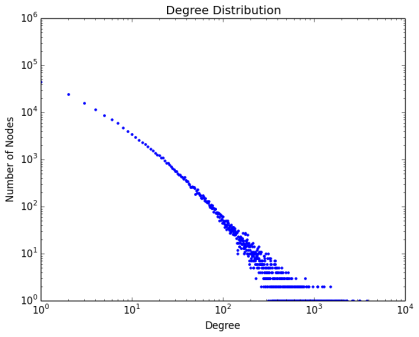

Figure 1: Degree Distribution of the Full Yelp Dataset

From this, we can see that the Yelp Dataset follows a Power Law degree distribution fairly well. This makes sense since with most social networks, there exhibits preferential attachment behavior. In the case of the Yelp Social network, this could be explained by the fact that the most active Yelp users likely post the most reviews on Yelp and as a result, their reviews are seen by more users. Additionally, the earliest Yelp users have also had more time to visit more restaurants and rate them. As a result, the users with the most ratings have the greatest opportunity to make friends with other users.

Though it would be great to work with such a large and dense dataset, it produces multiple challenges. The first, is that the dataset features businesses from specific cities in the world. As a result, the user friendship graph will likely exhibit strong clustering for those people who are from those cities. Trying to measure similarity between two nodes located in different continents doesn't make as much sense.

The second issue is that the dataset is too large to compute in a realistic amount of time given the tools and resources at our disposal. At 366K nodes and N^2 possible pairs of nodes, this problem isn't feasible with the given dataset.

3.2 Pruned Dataset: Pennsylvania Yelp Data

To address the problems with the entire Yelp Dataset, we decided to choose a specific state with a medium sized number of users to run our prediction algorithm on. This directly addresses the first issue of clustering since we simply users in the same state are more likely to be in the same friendship network. Additionally, the smaller size makes the problem of calculating similarities between pairs of nodes more feasible. We did this by taking only businesses located in Pennsylvania. We only took reviews on businesses from the subset of businesses and users who wrote those reviews.

After this process of pruning the dataset, we now have 2.9K users, 3.0K businesses, and 64.2K reviews. From the user friendship graph, we have 13.6K edges making for an average degree density of 4.5 edges per user.

The following is the degree distribution graph for the Pruned Yelp Dataset focusing on Pennsylvania:

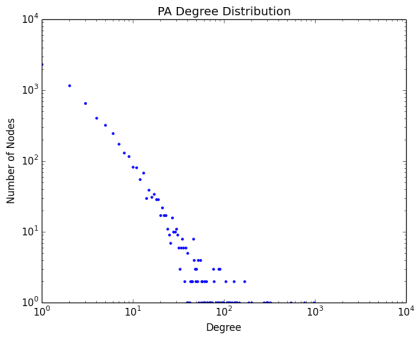

Figure 2: Degree Distribution of the Pruned Yelp Dataset for Pennsylvania

3.3 Representative Subset: Arizona Yelp Data

Our pruned dataset had the obvious drawback of being a very small subset of the entire Yelp population. We needed a more representative dataset to run our similarity measures on and as a result used the Arizona dataset as our larger dataset.

The Arizon dataset is the largest dataset of all states represented in the whole Yelp dataset. It's so large that we had to limit the number of businesses included to make calculations feasible, but the resulting dataset has many more users for the given businesses. It includes 17.7K users, 2.5K businesses, and 55.4K reviews. Though the number of businesses and reviews are lower, the number of users is an order of magnitude higher making the social data calculations much richer. The average degree density is 8.7 edges per user.

From the graph, we can see that the Arizona dataset is much more representative of the full Yelp dataset . It follows a power law distribution. Using this dataset, we can be confident that the results will more closely resemble the results of running our rating predictions on the full Yelp dataset.

4 Methodology

4.1 Predicting User Ratings with Recommender Systems

To go from a similarity metric to a prediction, we implemented a KNN Regressor according to the following equation:

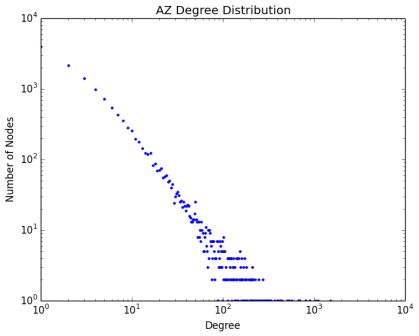

Figure 3: Degree Distribution of the Pruned Yelp Dataset for Pennsylvania.

Let r_x be the vector of user x ratings Let N be the k most similar users to x who have rated item i Prediction of item s for user x [6]:

$$r_{xi} = \frac{\sum_{y \in N} sim(x, y) \cdot r_{yi}}{\sum_{y \in N} sim(x, y)}$$

The way we will extend this research is by calculating user similarity based on properties of the user social graph instead of the above similarity measures.

4.2 PageRank Random Walk Similarity

One possible measure of similarity based on the user social graph is a random walk computation of similarity. By starting at a specific node and randomly walking from there, we get centrality value relative to that starting node. Random walk is a well studied algorithm that gives a stochastic description of of a graph structure taking into account neighbors of a node and the relationship those nodes have with each other. Nodes that are visited more often are likely to be more similar to the starting node because there's more paths between the nodes.

Using a very simply PageRank random walk algorithm of randomly walking nodes and counting occurrences of hits each node gets relative to a starting location, we likely arrive at the following simplified PageRank equations for each node:

$$R(u) = c \sum_{v \in B_u} \frac{R(v)}{N_v} \tag{1}$$

R is the PageRank of node u. c is a normalization factor. B is the set of nodes that are connected to u. N_u is the number of links from u.

4.3 Community Similarity

To soften the friends of friends limitation of the Jaccard method, we decided to expand our search for similar nodes to the community level. Using the standard Clique Percolation Method, we broke up our social network into communities formed by k-cliques. Similarity between nodes in the same community is set to 1, and 0 otherwise. Our hypothesis in this case was that the will of the community could influence and thus predict the rating behavior of Yelp users.

We know that business ratings don't occur in a vacuum. Users in the same community of a user could lead to influencing a user's rating accordingly. For example, members of a community may have visited a restaurant together and during the meal, complained terribly about the experience. As a result, if one member gives the restaurant a low rating, it's likely other users will give the restaurant a low rating as well.

4.4 Jaccard Similarity

Our final measure of similarity is based on the proportion overlap of friends for two given users, known as the Jaccard Index or Jaccard Coefficient [7]. The basis for this is that users with the most similar friend sets may also have a similar set of business reviews on Yelp. This is just another way to use the social network data to derive information about similarities between two users that aren't possible with the user-review information alone.

Given two users x and y, let F_x be the set of all nodes that have an edge to x and $F(y)$ be the set of all nodes that have an edge to y. The Jaccard similarity is given given by:

$$sim(x, y) = \frac{|F_x \cap F_y|}{|F_x \cup F_y|}$$

4.5 Centrality Vector Distance

Departing from the other measures, we decided to create a measure that did not at all depend on the edges connecting a user to potentially similar users. For this method, we hypothesized that the role a user plays within the network can be a predictor of its behavior. To quantify this "role" we used a vector of betweenness, closeness, and degree centrality. We postulated that perhaps users that serve the same function on a local level (e.g. the highest centrality users in their areas) act similarly despite potentially large separations from each other within the global network. For any pair of nodes, their similarity score was simply the cosine distance between their two centrality vectors.

5 Results

5.1 Evaluation

As discussed in [2], Herlocker et. al. point out that an appropriate accuracy metric that aligns with the goal of the recommendation is crucial to the success of the recommender system. One such accuracy metric that makes sense for the goals of our recommender system is the **Mean Squared Error (MSE)** since we don't care as much about small differences in predictions, but we care much more about large differences between predicted and actual ratings. The advantage of using MSE is that it's a relatively understandable metric since it is clear how error is being measured. The MSE is defined as follows:

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(\hat{Y}_i - Y_i)^2$$

where $\hat{Y}_i$ is the predicted value of an unknown rating, Y_i is the actual or observed value of that rating, and n is our number of predictions. A perfect system would have an MSE of 0. Table 1 displays the MSE for our different models.

As a second data point, we also compare the performance of our models using the Euclidean distance between the observed and predicted ratings (L_2 norm of the difference vector). The Euclidean distance is defined as follows:

$$L_2 = \sqrt{\sum_{i=1}^{n}(\hat{Y}_i - Y_i)^2}$$

Again, a perfect predictor here would have an L_2 of 0.

We were able to achieve major gains over our naive baseline of random guessing (69% reduction in MSE, 56% reduction in L_2. Our best result was a MSE of 0.950 and L_2 of 133.8 achieved by our Community Similarity measure. After some experimenting, we were able to achieve the best results for Community Similarity with a k of 5 (refer to use of k-cliques in Methodology). The success of the Community surprised us as our initial hypothesis was that the most local information would be the most useful in predicting an individuals behavior. However, it appears that observing action at the community

Regression Model	Similarity Measure	MSE	L_2	Fraction Predicted	Dataset
Random	N/A	3.107	304.4	1.00	PA
kNN	Jaccard	1.065	170.7	0.91	PA
kNN	Community	0.950	133.8	0.92	PA
kNN	PageRank	1.096	176.9	0.95	PA
kNN	Centrality	0.992	170.8	0.98	PA
kNN	Jaccard	1.282	192.8	0.91	AZ
kNN	PageRank	1.100	181.7	0.95	AZ

Table 1: Mean squared error, Euclidean distance results from various regressors and similarity measures. Also includes fractions of possible predictions actually made and the dataset being considered (Arizona or Pennsylvania).

level produces a good mix of local influences without becoming so broad as to merely predict the global average. As we can see from the confusion matrix for Community Similarity, our biggest errors come from predicting too high a rating for observed low ratings (e.g. for observed rating 1 and 2, we often predict 3 or 4). We attribute this error to the overall left skew of the ratings distribution (i.e. there are more ratings above 3 than below it). Thus, we may often encounter the situation where one member of the community rates a restaurant much lower than others in the community (we would expect a larger community to have an average rating closer to the global average). In this case, we are bound to predict too high a score. This could be remedied by precalculating how often negative outliers as described occur within a community.

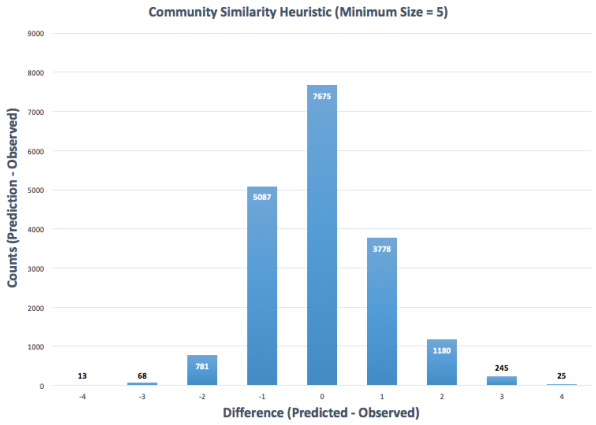

Figure 4: Histogram of errors (predicted - observed) for our Community Similarity (Minimum Community Size of 5).

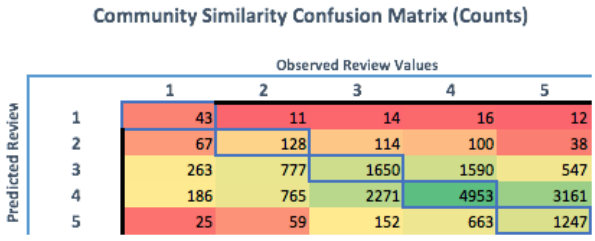

Figure 5: Confusion matrix for our Community Similarity Heuristic (Minimum Community Size of 5)

Our Centrality Vector Distance Similarity measure was also promising. This result helped us confirm that a user's position within its portion of the graph can be a good predictor of the behaviors of that user. The real potential of this method comes from its ability to be expanded. A major part of our future work involves expanding our user vector to include more information than centrality measures. Our hope is that more features, augmented by learned weights for each feature (assuming we use a Machine Learning model), would greatly improve our performance. We see the same trends in the confusion matrix for Centrality Vector Distance Similarity as we do for Community Similarity, we often predict too high.

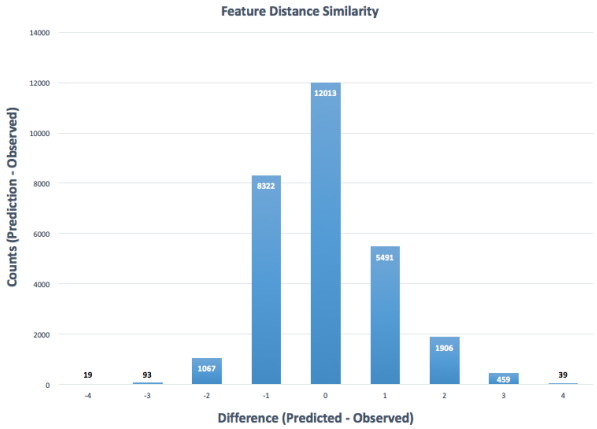

Figure 6: Histogram of errors (predicted-observed) for our Centrality Vector Distance Similarity.

Feature Distance Similarity Confusion Matrix (Counts)

Predicted Review \ Observed Review Values	1	2	3	4	5
1	71	21	26	32	19
2	111	160	179	152	61
3	494	1196	2306	2521	889
4	385	1221	3347	7575	5601
5	39	74	191	837	1901

Figure 7: Confusion matrix for our Centrality Vector Distance Similarity.

While running our model with different similarity measures, we realized that in some cases we were unable to actually predict a rating using kNN. For example, let there be some user A rating a business B we are trying to predict. Say all other raters of B have no friends in common with A. If we were using the Jaccard coefficient, this would produce a rating of 0, which is impossible. To improve our performance, we choose to ignore such cases where we had insufficient data to make a prediction. However, in order to compare different similarity measures, we keep track of how many predictions we had to discard due to insufficient data, and return the fraction of possible predictions our system was actually able to make. For instance, Community Similarity produced the most accurate predictions, but for ultimately fewer potential user-business pairs. We believe this is an important metric to note, a model that makes too few predictions has a diminishing value in a business application. All of our models achieved prediction rates of over 90%, which we believe to be an acceptable threshold.

Our Arizona dataset proved to be both an asset and a challenge. It was extremely promising to see our PageRank and Friendship Overlap similarity algorithms perform just as well on the larger and more representative Arizona dataset. It goes to show that our graph algorithms have relevant applications to real world social networks. We faced greater challenges running our Community and Feature Distance Similarity algorithms on the Arizona dataset. With these algorithms requiring $O(n^2)$ running time, the 10x increase in users proved be out of the scope that our machines could handle. Although not all of

our algorithms were feasible on this dataset, we thought it was extremely important to test on a data representative of real world networks in Yelp.

6 Further Work and Conclusion

In conclusion, we find the results of these predictions quite promising. Each predictor significantly surpassed the MSE of our baseline. This demonstrates that social network data is very relevant to making predictions about ratings. We find two major lessons from these results: 1. Communities of individuals tend to think alike. This makes sense as friends tend to follow groupthink and their ratings tend to be the same. 2. Similar types of people tend to think alike. Those who have many friends or tend to be leaders of opinion behave similar to one another since their behavior is likely correlated with their success in the social network.

We see a great deal of potential in this work as many facets of social predictions have yet to be explored. First, we would like to see how our algorithms apply to denser and more widely used social networks such as Facebook. The Yelp dataset is a great starting point, but in reality isn't what people would consider a social network by today's standards. Additionally, we hope to push the limits of our current algorithms and technology by finding ways to make our more costly algorithms scale. Our two most successful similarity algorithms also take the longest to run because of the nature of the algorithm. Finding ways to distribute the computations may prove to be a huge breakthrough in applying our work to larger datasets and make predictions even more generally.

References

[1] Anderson, A., Huttenlocher, D., Kleinberg, J., & Leskovec, J. (2012). Effects of user similarity in social media. *In Proceedings of the fifth ACM international conference on Web search and data mining*, (703-712).

[2] Herlocker, J. L., Konstan, J. A., Terveen, L. G., & Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems (TOIS)* **22(1)**, 5-53.

[3] Fouss, F., Pirotte, A., Renders, J. M., & Saerens, M. (2007). Random-walk computation of similarities between nodes of a graph with application to collaborative recommendation. Knowledge and data engineering *IEEE transactions on*, **19(3)**, 355-369.

[4] Chevalier, J, and Mayzlin, D (2006). The effect of word of mouth on sales: Online book reviews. *Journal of marketing research* **43.3**, 345-354. APA

[5] Leskovec, J, Huttenlocher, D, and Kleinberg, J (2010). Predicting positive and negative links in online social networks. *Proceedings of the 19th international conference on World wide web.* ACM.

[6] Leskovec, Jure. "Recommender Systems and Collaborative Filtering." (n.d.): n. pag. Stanford CS 246. Web. 15 Oct. 2015.

[7] Tan, Steinbach, and Kumar. "Data Mining: Data." University of Minnesota Computer Science, n.d. Web. 16 Nov. 2015.