Analysis and Prediction in the Yelp Network

Dilsher Ahmed, Miguel Camacho-Horvitz, Raunak Kasera

Computer Science Department, Stanford University

1 Introduction

Link prediction and the understanding of social influence propagation have many useful applications in modern networks. This understanding can be applied to provide recommendations to users (e.g. music recommender platforms or social friend/contact/follower suggestions) as well as to predict the nature of links in the graph (e.g. friendship or enmity). These topics have been studied on various large networks such as Wikipedia and Facebook. With the yearly Yelp Dataset Challenge, data scientists have the opportunity to tackle these same challenges on the dynamically growing Yelp network.

The Yelp network's combination of a bipartite review graph and a friendship graph between users provides an opportunity to leverage network properties on real data using existing algorithms. Previous papers have attempted link prediction on bipartite graphs, and we will seek to discover whether social influence can be used to further improve such predictions while also determining whether social influence indicators play a large role in the Yelp network. This will allow for a better understanding of the role of social influence in real-life bipartite graphs while also providing a method for user-specific recommendations and targeted businesses advertising.

2 Related work

Link prediction has been thoroughly studied in dynamic network analysis. Liben-Nowell and Kleinberg [5] studied a variety of different methods for link prediction including 'Node Neighborhood Based Methods' from which we drew heavily. In a paper from last year's projects [4], the authors' primary focus is to gauge the success rate of various algorithms at predicting the existence of links in the bipartite Yelp network. Given our work also focuses on Yelp's network, we were able to expand on their work. The authors used several algorithms and we have adapted some of the algorithms and testing mechanisms.

Social influence has also been studied in many contexts. In [1] and [2], the authors sought to analyze how social influence plays a role in influencing people's opinions. In both, an artificial scenario was created and users were asked to rate some item with the absence and presence of a stimulus. It was determined that the presence of the stimulus (in one case the number of downloads for the song being rated and in the other an approval rating for a comment being rated) did indeed influence new reviewers. We hypothesized that reviewers in Yelp would be similarly influenced. In [3], the authors primarily focus on the identification of early adopters in a social network and determining which early adopters have maximal influence. Additionally, they tried to incentivize such adopters to improve potential opportunities for marketing. We intend to determine if incorporating the influence of users can help in a collaborative filtering model for link prediction.

3 Data

3.1 Dataset and Preprocessing

Our project test bed is the Yelp dataset available from the 8th round of the Yelp Dataset Challenge. This dataset comprises of 2.7 million reviews by 687,000 users for 86,000 businesses in six American and four international cities and also has an attached social networking graph relating users to one another. We filtered the cities to remove those cities with a low number of reviews and grouped other cities together if they shared a metropolitan area e.g. Las Vegas and Henderson. This grouping was done to prevent sparse graphs from forming. After this grouping, we were left with 6 major cities: Las Vegas, Phoenix, Madison, Montreal, Pittsburgh and Charlotte, each with over 50,000 reviews.

We formed an undirected bipartite graph for each city with the two sets of nodes being users and businesses with an edge between a user and a business if the user reviewed the business.

3.2 Initial Findings

For each city, our basic statistics are as follows:

City	Users	Businesses	Reviews	Reviews/User	Reviews/Business
Las Vegas	330348	22473	1097471	3.32	48.8
Phoenix	218057	28160	815484	3.74	29.0
Charlotte	43174	5695	143710	3.33	25.2
Pittsburgh	30841	3628	101501	3.29	28.0
Montreal	23853	4467	78108	3.27	17.5
Madison	17797	2278	56526	3.18	24.8

Table 1 - Basic statistics for all cities in our dataset

There are two primary outliers: Montreal and Las Vegas. Montreal has a relatively low average degree per business. This may be because Montreal is the second largest city in Canada giving a reason for a larger number of businesses to target it to reach a new market over targeting similar-sized cities in the United States. Las Vegas' outlier status follows from Las Vegas being a tourist destination with a large influx of people passing through and not many people taking up residence there. Thus, businesses in Las Vegas tend to have an average degree of 50 but users tend to have a lower average degree when compared to Phoenix. This behavior can also be seen in Figure 2 by observing the relations of the curves for Las Vegas and Montreal to other cities.

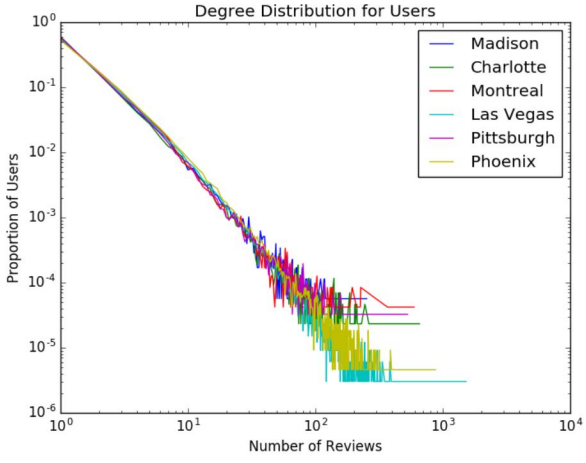

Figure 1 - Number of Reviews by Users in our Cities

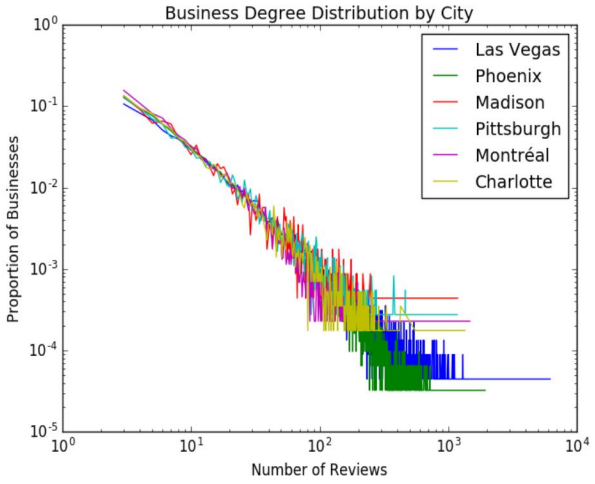

Figure 2 - Number of Reviews of Businesses in our Cities

4 Link Prediction

The first method used to analyze the behavior of social influence was link prediction. We performed link prediction using two different datasets for each city and used a variety of different algorithms detailed in the following sections. Our overall outline was to:

1. Generate our training and test datasets for every city.
2. Perform link predictions with various thresholds and algorithms.
3. Evaluate all our results against the appropriate test dataset.

4.1 Dataset Generation

We have two primary metrics for generating datasets for every city: a time-based metric (where we filter out reviews in the last year) and a random metric (where we filter out reviews randomly) adapted from [4]. Both metrics initially generate test and training datasets of the same size but we take the largest weakly connected component of each.

City	Training Data Size (Time)	Training Data Size (Random)	Test Data Size (Time)	Test Data Size (Random)
Phoenix	212625, 662337	196120, 523094	140312, 290310	138796, 286693
Pittsburgh	29826, 82724	26981, 65754	17257, 35482	18925, 34960
Madison	17465, 46758	16028, 38236	9303, 18059	9995, 17662
Charlotte	41610, 115990	37460, 91271	25696, 51904	26373, 51028
Montreal	24587, 62057	22031, 47907	14624, 29963	16642, 29558

Table 2 - This table shows (Number of Nodes, Number of Edges) for every dataset

The random metric sometimes ends up with smaller training sets owing to the graph becoming slightly disconnected when generating the two datasets.

4.2 Implementation

Link prediction is performed by looping over all pairs (u, b) where u is a user and b is a business and generating a rating for each pair based on some algorithm.

4.2.1 Defining neighborhoods

In order to explain link prediction, it is necessary to understand the definition of the neighborhood of a node as defined in [4].

The neighborhood of a user U is written as N(U) such that:

N(U) = The set of all users who have reviewed a business B that has been reviewed by U

The neighborhood of a business B is written as N(B) such that:

N(B) = The set of all users who have reviewed business B.

4.2.2 Link Prediction Rating Algorithms

The 3 link prediction algorithms we used are:

1. Delta
2. Preferential-Attachment
3. Random Baseline

Delta was chosen as it performed best in [4]. Preferential-attachment was chosen as we theorized that highly rated business tend to get reviewed more often hence mimicking the "rich-get-richer" phenomenon. And the random algorithm was chosen as a baseline.

Delta:

$$rating(U,B) = \sum_{v \in N(U) \cap N(B)} \frac{2}{|N(v)| * (|N(v)| - 1)}$$

Preferential-Attachment:

$$rating(U,B) = |N(U)| \cdot |N(B)|$$

Random:

$$rating(U,B) = random\ between\ [0,1)$$

4.2.3 Link Prediction Algorithm

The formal algorithm for link prediction is as follows:

Let (V_{Tr}, E_{Tr}) be the vertices and edges in our training dataset and let (V_{Te}, E_{Te}) be the vertices and edge in our test dataset and let *thr* represent the value of the threshold where *thr* $\in \{1, 10, 25, 50, 65, 75, 90, 100\}$.

1. For every user $U \in V_{Tr}$, generate N(U) and for every business $B \in V_{Tr}$, generate N(B)
2. For each pair $(U, B) \in V_{Tr} \times V_{Tr}$ where U is a user and B is a business:
    1. If $(U, B) \in E_{Tr}$, rating(U, B) = 0
    2. If $|N(U) \cap N(B)| <$ threshold, rating(U, B) = 0
    3. Otherwise compute rating(U, B) according to each algorithm outlined in 4.2.2
3. Select the $|E_{Te}|$ pairs with the highest rating and evaluate the precision of each of our three algorithms.

We performed this test over all values of the threshold and for both our generation metrics.

4.3 Results and Evaluation

We evaluate our results by computing the precision i.e. the percentage of the $|E_{Te}|$ predicted links that were actually present in our test dataset. Despite Delta performing well in [4], our initial results have demonstrated that a preferential-attachment model tends to perform the best on smaller cities particularly when the datasets are generated based on time. Preferential attachment tends to have precision around 1.5% at its peak for a small city whereas Delta tends to peak at around 1.3% precision. On larger cities such as Phoenix, Delta tends to outperform Preferential Attachment with precision 0.08% to 0.05%.

With a randomized dataset, Delta manages between 6-8% precision on all cities whereas Preferential Attachment manages between 4-7% precision and tends to be outperformed by Delta. Our baseline algorithms perform worse than both Delta and Preferential Attachment in every case. The superior performance of Preferential-Attachment on time-based filtering does indicate that users do tend to flock towards popular businesses confirming our initial hypothesis that the "rich-get-richer" phenomenon is indeed observed in the Yelp graph. One of the reasons for the lower precision among time-based filtering is that it never predicts links to businesses that opened within the last year thereby performing somewhat worse.

Note that these precision values are extremely low owing to the nature of our problem as we predict an extremely large volume of links. Therefore precision@K tends to be a better measure of our algorithm's performance. We evaluated precision@K for every city and the graphs looked similar. A sample graph for Pittsburgh is shown below.

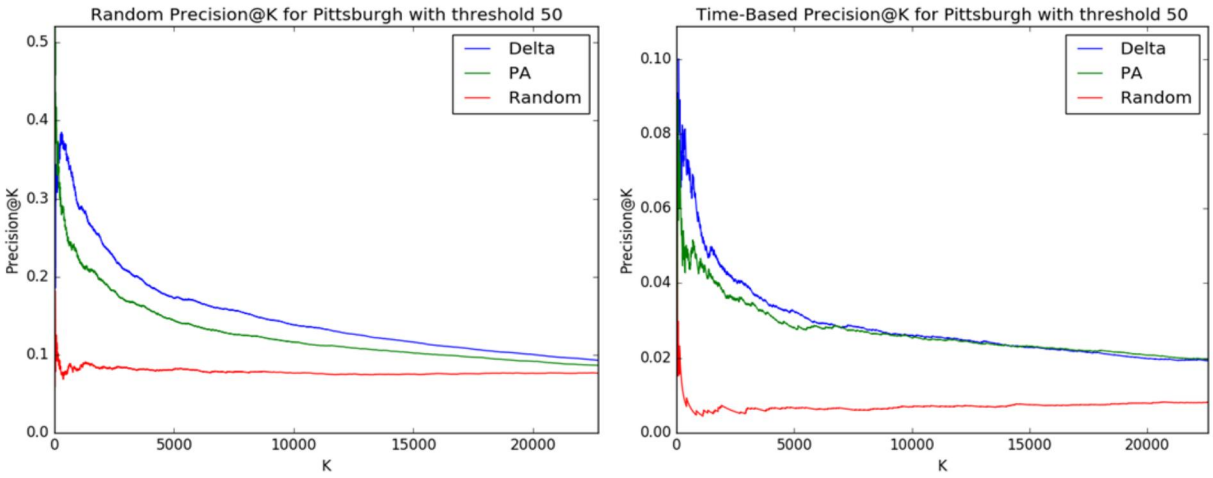

Figure 3 – An evaluation of our algorithms' performance on both datasets

As seen in Figure 3, the random baseline algorithms tend to have constant Precision@K for both testing models. The precision@K for Delta tends to be the highest and the difference between Precision@K for low values of K indicates that our prediction metrics are relatively accurate on the first 5000 or so predictions. For time-based analyses, our algorithms tend to outperform the baseline by a factor of 4 initially and by a factor of 2 at $K = 5000$. It can also be noticed that time-based predictions tend to have Precision@K values which are roughly $1/5^{th}$ of the corresponding values for the randomized datasets. On randomized datasets, our algorithms tend to greatly outperform the random baseline as they have 4 times as high Precision@K even for large values of K such as 20000.

5 Social Influence

Next, we analyzed how social influence behaves in the Yelp network. We attempted to perform this analysis by considering stimuli such as previous ratings and the number of ratings and determining how these impacted the ratings given by future users.

As our intuition suggested and the results in [1] found in their experimentation, we hypothesized that Yelp businesses with established reputations would draw further customers and hence receive both more as well as better reviews. To test our hypothesis we examined the average ratings given to businesses grouped by the number of reviews these businesses had. Additionally, we examined the variance in the ratings given to business early on versus later on as well as the variance in ratings given to businesses of high repute versus low repute.

To determine whether social influence indicators such as the number of reviews played a major role in determining the rating of a business, we first classified businesses into three distinct categories based on the number of reviews. Businesses with fewer than 50 reviews received the 'few reviews' label, while businesses with between 50 and 100 reviews received the 'medium reviews' label and businesses with over 100 reviews received the 'many reviews'

label. For all the cities, between 80-90% of the businesses had few reviews with roughly 5-8% falling into each of the other two categories, resembling a power-law distribution.

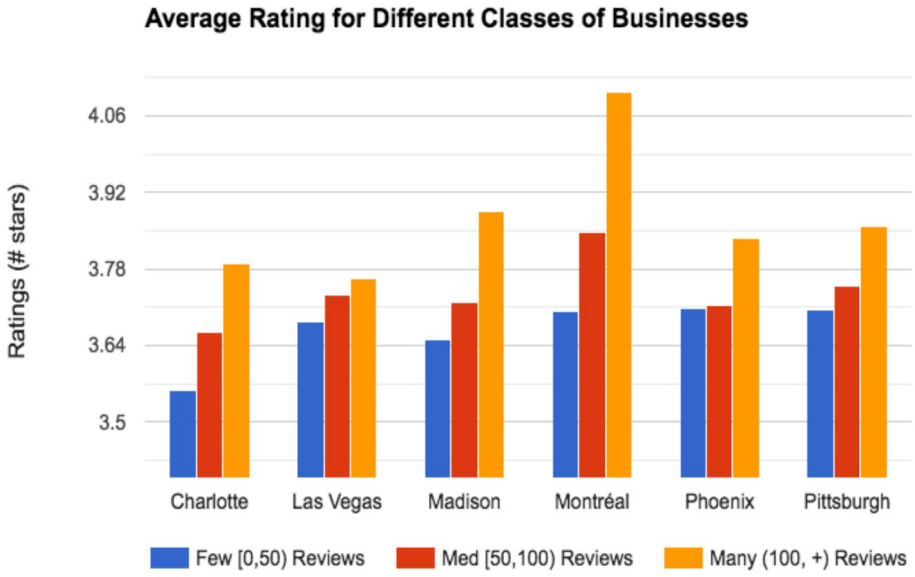

Figure 4 – The mean rating for businesses from each city grouped on the number of reviews

Our initial results (Figure 4) clearly confirmed the correlation between number of reviews of a business and its rating, as in all cities we found a monotonic increase in rating over our stratified number of reviews labels.

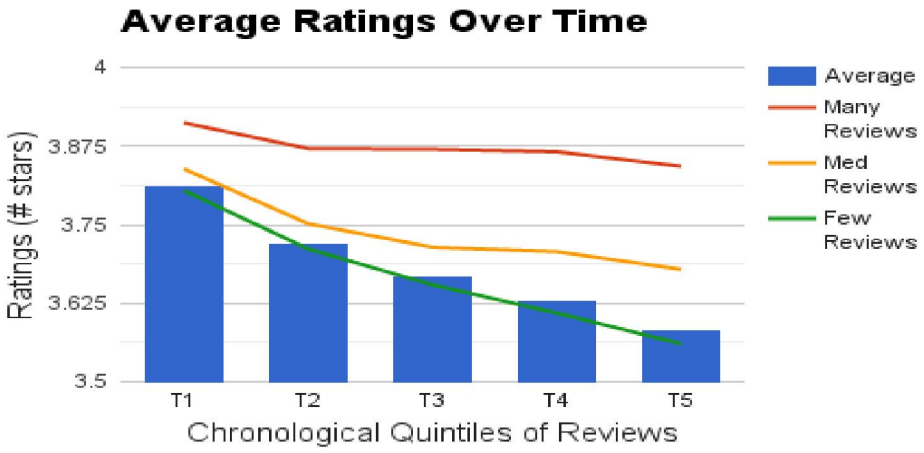

Figure 5 – The mean rating for businesses in all cities grouped on the labels and then by timestamp.

Interestingly, some of our further results contradicted our early hypotheses. Further tests, which produced results such as shown above in Figure 5, confirmed that greater-reviewed businesses do better, yet also showed that no matter the reviews of the business there was a strict decline in rating over time. Our refined hypothesis is that this is the result of 'expectations'. Given that the average business starts with a rating over 3.8, users come in with expectations that may or may not be met. While initially there may have been a 'surprise' at the business' product given the lack of reviews, a product that has been highly rated is subject to more critique and can lead to user disappointment.

Finally, to test the variance in ratings assigned to businesses over time - especially to test our hypothesis that there would be more variation initially - we computed the standard deviations observed in popular and unpopular businesses among the first and last quintiles of their reviews.

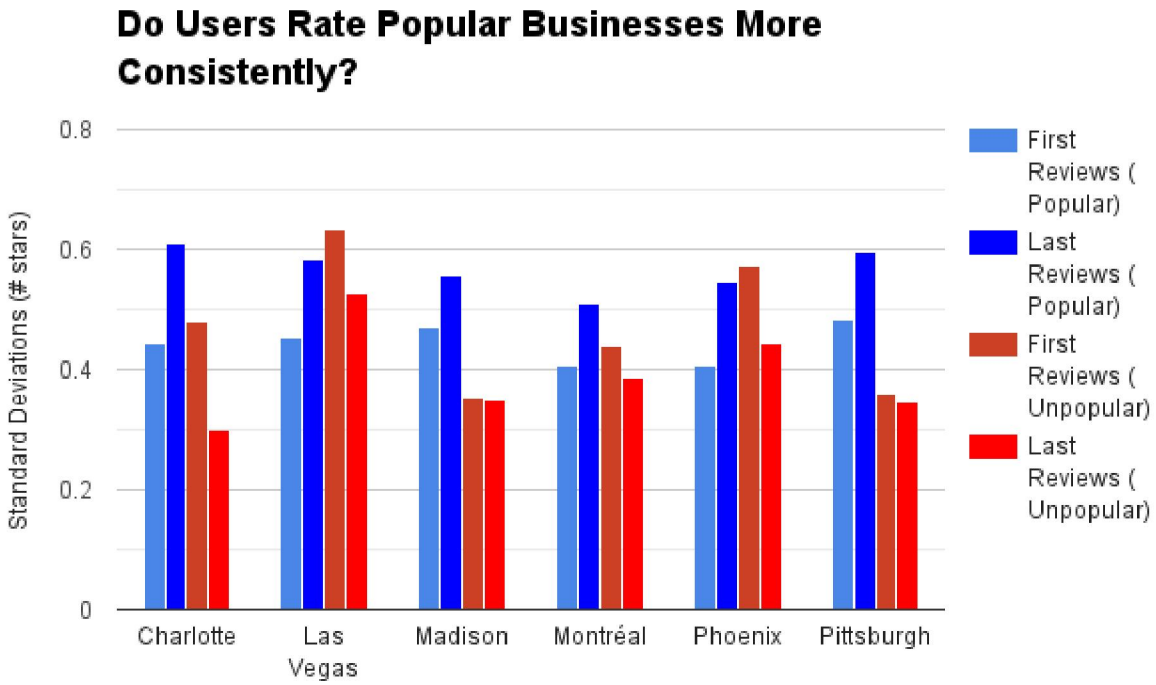

Figure 6 – The standard deviation for initial and final reviews for businesses grouped based on popularity

What we found turned out to partly contradict our initial hypothesis. While our initial hypothesis may have been true for unpopular businesses (where we defined unpopular as having an overall rating between 1 and 2 stars), among popular businesses (between 4 and 5 stars), the standard deviation grew between the initial and final reviews in all of our cities. Our refined hypothesis is now that among businesses with many reviews, which correlate strongly with positively-rated businesses, the business's launch generates a certain hype and gains positive momentum as early positive ratings could be evidence of. Then, as aforementioned, over time as a business develops a positive reputation, people may be more likely to be 'let down' as they bring high expectations rather than being 'positively surprised' as with a new business. Thus, they may receive more mixed reviews later on. Among businesses with less reviews, there is less expectation coming in which can explain why there is less of a difference in initial versus final rating variance.

6 Collaborative Filtering

The Yelp dataset is not restricted to a bipartite graph mapping users to businesses via reviews as it also includes a social network graph representing friendships among various users. The existence of this graph has not been leveraged as much in previous work and in order to use and evaluate the usefulness of this data, we performed two separate instances of collaborative filtering which utilized various features including the structure of the friendship graph to compute our similarity matrices. For both instances of collaborative filtering, we generate a training and test dataset by filtering out the reviews within the last year as our test dataset.

6.1 Collaborative Filtering on Businesses

We performed item-based (in this case business-based) collaborative filtering, generating a similarity matrix for businesses within the same city. The similarity for a pair of businesses is computed by incorporating various features such as the average star rating, the overlap between the sets of reviewers for each business and the number of reviews. We experimented with various normalization techniques and various weights and found that all our results tended to be very similar with a difference of at most 1% between results. The final similarity score between businesses b_1 and b_2 was computed as follows:

$$sim(b_1, b_2) = \frac{|N(b_1) \cap N(b_2)|}{|N(b_1) \cup N(b_2)|} + \left(1 - \frac{\left|\log\left(r(b_1)\right) - \log\left(r(b_2)\right)\right|}{maxr}\right) + \left(1 - \frac{|s(b_1) - s(b_2)|}{4}\right)$$

In this equation $r(b_1)$ represents the number of reviews of b_1, maxr represents the maximum number of reviews for any business and $s(b_1)$ represents the average rating for b_1. We predicted 5 businesses for every user and evaluated precision by counting the number of correct predictions made i.e. the times we predicted that a user would visit a restaurant which they ended up visiting. This was divided by the total number of predictions made. Note that for users with degree d lower than 5 in the test dataset, we only evaluate the first d predictions.

City	# of Correctly Predicted Links	# of Total Predicted Links	Precision
Madison	2855	11269	25.33%
Montreal	3971	17162	23.14%
Pittsburgh	5890	21125	27.88%
Charlotte	8197	31559	25.97%

Table 3 – Results of Item-Based Collaborative Filtering

The precision of our measurements tended to be approximately 25-26% with Montreal having the lowest precision which may be due to the fact that the average degree of a business in Montreal is lower than the average degree of a business in other cities. This relationship appears to be relevant particularly as precision tends to be higher for cities with higher average degree per business. This algorithm usefully provides a way for businesses to perform targeted advertising to users who are likely to review them in the future. The baseline measurements for the collaborative filtering is extremely low and is on the order of 10^{-4}. Note that this test was not performed on Phoenix and Las Vegas owing to the extremely large RAM requirement for computing the similarity matrix and keeping it in memory.

6.2 Collaborative Filtering on Users

We performed user-based collaborative filtering, generating a similarity matrix for users within the same city. The similarity for a pair of users is computed by incorporating various features such as the average star rating given by each user, the number of reviews by each user, the overlap between the sets of friends of both users and the overlap between the businesses reviewed by each user. We experimented with various normalization techniques and various weights and found that our results tended to be very similar as occurred for businesses. The final similarity score between businesses u_1 and u_2 was computed as follows:

$$sim(u_1, u_2) = \frac{|N(u_1) \cap N(u_2)|}{|N(u_1) \cup N(u_2)|} + \left(1 - \frac{\left|\log\left(r(u_1)\right) - \log\left(r(u_2)\right)\right|}{maxr}\right) + \left(1 - \frac{|s(u_1) - s(u_2)|}{4}\right) + \frac{|F(u_1) \cap F(u_2)|}{|F(u_1) \cup F(u_2)|}$$

In this equation $r(u_1)$ represents the number of reviews by u_1, maxr represents the maximum number of reviews by any user, $s(u_1)$ represents the average rating for u_1 and $F(u_1)$ represents the set of friends of u_1. We predicted 5 users for every business and evaluated precision by counting the number of correct predictions made i.e. the times we predicted that a user would visit a restaurant which they ended up visiting. This was divided by the total number of predictions made. Note that for businesses with degree d lower than 5 in the test dataset, we only evaluate the first d predictions.

City	# of Correctly Predicted Links	# of Total Predicted Links	Precision
Madison	1005	5730	17.54%
Montreal	2431	11648	20.87%
Pittsburgh	1885	9827	19.18%
Charlotte	2667	14869	17.94%

Table 4 – Results of User-Based Collaborative Filtering

The precision of our measurements varies between 17 and 21%. One of the major factors that leads to this precision being low is the relative sparseness of the friendship graph. For all 4 of our cities at least 55% of the users have no friends on Yelp. Similarly, the number of fans for each user, which was another feature we initially considered, also has over 75% of users having 0 or 1 fans. This algorithm provides a useful way to improve user experience by recommending businesses that the user would be likely to review in the future. The baseline measurements for the collaborative filtering is extremely low and is on the order of 10^{-4}. Note that this test was not performed on Phoenix and Las Vegas owing to the extremely large RAM requirement for computing the similarity matrix and keeping it in memory.

7 Conclusions

In this paper we explored the behavior of social influence in the Yelp network i.e. determining how users' behavior is influenced by considering previous ratings as stimuli. We also analyzed various link prediction algorithms on our dataset and attempted to utilize the friendship graph to improve upon collaborative filtering methods.

- Our results appeared to support the hypotheses outlined in [1] and [2] that indicators such as the number of reviews affect users' behavior as we saw that more reviews strongly correlated with higher ratings.

- There seems to be a correlation between the rating of a business and user behavior with user ratings tending to fluctuate highly for newly-opened unpopular businesses or established popular businesses.
- As stated in [4], link prediction algorithms are computationally and memory-intensive. The precision afforded by such algorithms is low but still better than our baseline.
- Even basic collaborative filtering tends to perform reasonably well at making predictions. The larger amount of data for businesses versus users means that item-item collaborative filtering tends to perform better than user-user collaborative filtering in their corresponding tasks. The sparseness of the friendship graph in particular caused difficulties.

8 Future Work

Although precision for our link prediction algorithms were low, these algorithms were performing an extremely large number of predictions with relatively sparse datasets. It would be interesting to combine the algorithms used with machine-learning techniques in order to improve upon this precision. Additionally, the friendship graphs were extremely sparse and so were not extremely helpful in our user-based collaborative filtering. With Yelp's growing user-base and increasing use as a social network, our predictions may improve on future datasets.

References

[1] M. Salganik, P. Dodds and D. Watts, "Experimental study of inequality and unpredictability in an artificial cultural market," *Science,* vol. 5762, no. 311, pp. 854-856, 2006.

[2] L. Muchnik, A. Sinan and T. S. J, "Social Influence Bias: A Randomized Experiment," *Science,* vol. 341, no. 6146, pp. 647-651, 2013.

[3] Y. Singer, "How to win friends and influence people, truthfully: influence maximization mechanisms for social networks," in *WSDM '12 Proceedings of the fifth ACM international conference on Web search and data mining* , New York, 2012.

[4] K. Mao, X. Cai and Y. Wang, "Link Prediction in Bipartite Networks - Predicting Yelp Reviews," 2015.

[5] D. Liben-Nowell and J. Kleinberg, "The link-prediction problem for social networks," *Journal of the American Society for Information Science and Technology,* vol. 58, 2007.

Contributions:

Dilsher Ahmed: Data cleaning, link prediction algorithm implementations, collaborative filtering implementations, generating dataset statistics, writing up the report and drafting the poster

Miguel Camacho-Horvitz: Preliminary data collection and analysis, social influence propagation analysis and drafting the poster, writing up the report and tabulating final results

Raunak Kasera: Preliminary data collection and analysis, problem formulation, running tests, tabulating final results, writing up the report and drafting the poster