Learnings From Yelp Network Properties

Derek Lim
Stanford University
limderek@stanford.edu

I. Abstract

New online evaluation networks come as results of social media networks. Ebay, Amazon, Stack Overflow, and Yelp are all examples of online networks where users submit their evaluation of a particular item whether it be another user, a product, etc. These networks allow a user to submit their opinion to be read and evaluated by other users in the network. These crowd-sourced reviews act as a method for users to infer evaluations like whether a restaurant is worth going to, if a product is good quality or whether to trust an online seller. In particular, we will discuss these kinds of online networks as a network where links correspond to a rating between two nodes.

We will look at the Yelp network specifically. A common model for the Yelp network is a bipartite graph. Instead we will take a novel approach of analyzing the network as triads consisting of 1 user and 2 businesses. From this proposed model, we can draw new conclusions about the underlying network structure. We analyze how status theory from the social sciences can explain the hidden principles of user evaluation. With these insights from the status theory we develop our proposed model to predict the value of links within the network. The work in this paper can be extended to several applications beyond Yelp. In general, it shows how a user's evaluation depends on the context or the network structure it is within.

II. Introduction

Social media networks have given people a new factor to consider when making everyday decisions. For example, before buying a new TV or furniture set, people can now first check its reviews on Amazon. When deciding where to eat pizza tonight, hungry customers can look up restaurant reviews on Yelp. Generally, these social networks provide an aggregated, crowd-sourced evaluation of a particular item.

Prior work has been done on predicting a Yelp review rating from sentiment analysis on the review text as well as predicting signed links in other social networks like Slashdot and Wikipedia from network characteristics [2]. This paper will take a network approach to predicting Yelp review ratings. In the Yelp network, users review businesses (mainly restaurants). A review comprises of a star rating (1 to 5) and a review text detailing their experience. We will explore the underlying network characteristics of the Yelp network and how that can be analyzed to draw conclusions about a given link. In particular we will answer the questions of: what information can be extracted from the network? What are the most common configuration of such links? For a given link, how can its value be inferred from its neighboring links?

III. Related Work

Prior work has been done on predicting positive and negative links in a signed social network. In the Epinion, Slashdot, and Wikipedia networks, the links can be connected to the theories of balance and status [2].

Recent work has been done on how relative status can contribute to a user's evaluation. That is in the sense, if User A is evaluating User B how does their status quo affect the way which User A will evaluate User B. Additionally, they also consider how similarity affects the evaluation. If User A and User B

are very similar, a positive evaluation is more likely [1].

A big obstacle in data science is obtaining quality real world data. Since Yelp releases a dataset to the public, researchers have analyzed the dataset from a number of perspectives. For example, the star rating can be predicted from sentiment analysis on the review text or from machine learning on a number of key user and business features.

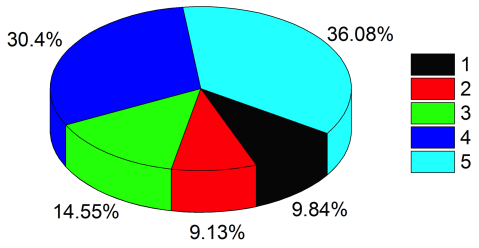
An important note is that depending on the nature of the network, different modeling approaches are typically chosen. In the case of Wikipedia adminship elections, users can be evaluated as well as give evaluations, so one approach is to look at the similarity between the two users [1].This model would not make sense for a user evaluating a product since the similarity metric would be invalid. In this scenario, a bipartite graph is commonly used. With a group of nodes representing the users or people giving the evaluation, and another set of nodes representing the product being evaluated. Directed links go from the user giving the evaluation to the product, and the sign would correlate to a positive or negative evaluation.

We propose a new model for the Yelp network which has not been considered. For each user, our proposed model looks at triads which consist of one user node and two business nodes. The links from user to business represent the value of the review rating. The link from business to business represent the relative status difference between the two and is computed from the average business rating.

IV. Dataset Description

In this paper, we study the Yelp social network. The data comes from the Yelp Dataset Challenge sponsored by Yelp. It consists of 42,153 businesses, 320,002 business attributes, 252,898 users, and 1,125,458 reviews. As shown in the figure, the dataset is comprised of the following: 110772 1 star reviews, 102737 2 star reviews, 163761 3 star reviews, 342143 4 star reviews, 406043 5 star reviews. We can see that the dataset is skewed towards positive reviews. For this reason, in the preliminary analysis we take 1, 2, and 3 star reviews as positive and 4, and 5 star reviews as positive to balance the dataset. Although representing 4 stars as negative would balance positive and negative links more, we do not do so since intuitively a 4 star rating is positive. Users can rate businesses on full integer star ratings (1, 2, 3, 4, 5) while average star ratings on businesses are defined at a half star granularity (1.0, 1.5, 2.0, 2.5....5).

Distribution of Star Ratings for One Million Yelp Reviews



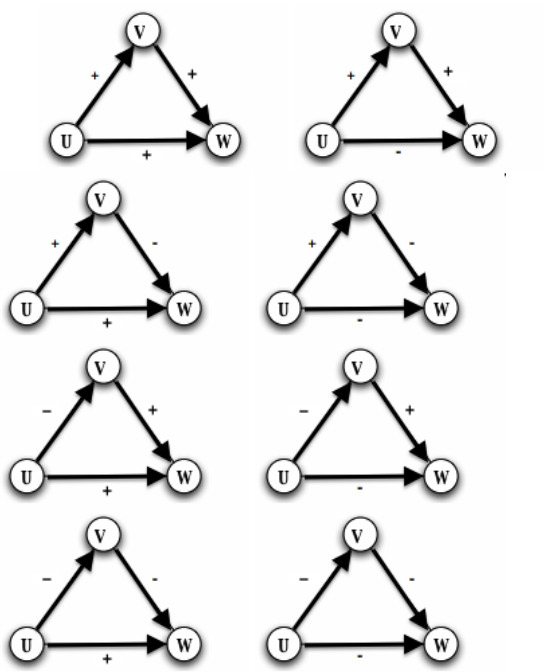
V. Preliminary Triad Analysis

The Yelp network is commonly looked at as a bipartite graph. We will take the novel approach of analyzing the network from a triadic standpoint consisting of 1 user and 2 businesses. We will define the network as a directed graph $G = (V, E)$ with V nodes and E links. Nodes are both users and businesses. Directed links go from users to businesses and from businesses to businesses. We do not consider links between users to users, and future work could include such links.

We will first consider the case where the edges are signed (positive, negative). Then we will then extend the model to the case where edges can take on additional values, namely 1 to 5.

We will use the following notation to represent a triad u, v, w consisting of 1 user u and 2 businesses v, w. Links between users and businesses only exist if the user has rated the given business before, otherwise there is no link. Links in the triad are defined from $u \to v$, $v \to w$, $u \to w$. Each link can take on two values (positive or negative), and there are 3 links, resulting in $2 * 2 * 2 = 8$ possible triad

configurations. The following notation will represent the signs of a triad: $(u, v, w) = PNP$ is a triad with a positive link $u \to v$, negative link $v \to w$, and positive link $u \to w$.
The sign between nodes u and v is represented as $s(u, v)$. The sign will be positive, negative, or 0 if no link exists. The sign of a link between a user and business represents whether the user gave the business a positive (4 or 5 stars) or negative (1, 2, or 3 stars) rating. That is $s(u, v) = 1$ for a star rating of 4 or 5, and $s(u, v) = -1$ for a star rating of 1, 2, or 3. The sign of a link between a business to business represents the relative status difference between the destination and source node. Status is defined by the average business star rating. Say business v has an average rating of 4.5 and business w has an average rating of 3.5, $s(v, w) = -1$. Additionally if v has a rating of 2.5 and w has a rating of 3, $s(v, w) = 1$.



We begin by analyzing a random 10,000 triads out of the dataset. We do this by first picking a random review. For that given, review, we make the user u and the business w. Given this user u, we pick another random review user u has made and making that business v. If the user has not rated two businesses then we can not make a triad out of this user. We continue this process until we have 10,000 triads. The frequency of the 8 possible triad configurations within the Yelp network are the following: $(u, v, w) = PPP$: 2226, $(u, v, w) = PNP$: 2010, $(u, v, w) = NPP$: 1604, $(u, v, w) = PNN$: 1552, $(u, v, w) = PPN$: 812, $(u, v, w) = NNP$: 701. $(u, v, w) = NNN$: 690, $(u, v, w) = NPN$: 405, Intuitively these results make sense given that the data is positively skewed. Since v and w are chosen at random, we see that both PPP and PNP are the most commonly seen in the dataset. Then likewise we have triads NPP and PNN where the sign of the edge $v \to w$ directly matches the sign of $u \to w$. From this, we infer that the status of w does not matter, but rather its relative status to v (another business u has reviewed) determines how u will rate it.

VI. Models

We will explore two main models: collaborative filtering and the triad model. **Collaborative filtering**. Given a user and a restaurant, first we find a group of similar users to the given user. We will define this group as the similarity group. Similarity is calculated from user features like average rating given to different categories of restaurants. Then based on this similarity group, we will make a prediction of what the given user will rate the given restaurant. We find the similarity group by looking at all the users who have gone to that given restaurant, and rank them based on similarity as defined later. From this similarity ranking we base our prediction on the top N most similar users as defined by our model.

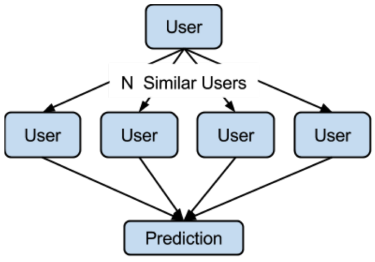


Figure: Finding N Similar Users

Utility Matrix. We define user features as the average rating given to different categories of restaurants. Example: Mediterranean: 2, Japanese: 4, Mexican: 4, Thai: 1. We will now define a utility matrix to capture these user features. We have two groups: users and user features. The rows of the matrix correspond to the users and the columns correspond to the user features. The value of a particular cell in row i and column j represents the rating which user i gave category j.

	Thai	American	Chinese	Italian
User 1	4		5	
User 2	2		2	4
User 3		4	4	

Figure: Utility Matrix

$$\text{sim}(\boldsymbol{x}, \boldsymbol{y}) = \cos(\boldsymbol{r_x}, \boldsymbol{r_y}) = \frac{r_x \cdot r_y}{||r_x|| \cdot ||r_y||}$$

Figure: Cosine Similarity

Similarity Measure. We explore three different methods of measuring similarity: Jaccard Similarity, Cosine Similarity, Pearson Correlation Coefficient. The flaw with Jaccard Similarity is that it ignores the value of the rating and only looks at the set of features with ratings. Pearson Correlation Coefficient is similar to Cosine Similarity but is slightly more computationally expensive. For these reasons, we settle on using Cosine Similarity.
To address the issue of all features being counted as positively correlated, we now normalize the utility matrix by subtracting the row mean from each value. Otherwise, if user 1 rates Thai restaurants 4 stars and user 2 rates Thai restaurants 1 star, this will result in a postive similairty when in fact it isn't.

Rating Prediction Methods. Now to arrive at a prediction we analyze two different methods, namely **unweighted average** and **weighted average**. For an unweighted average, we take the N most similar users, and do an unweighted average of those user's ratings of the given restaurant.
For a weighted average, we weight each of the top N most similar users by their similarity value. We compute the weighted average as defined in the figure. In the weighted average figure, let N represent the set of N most similar users, s_{xy} represent the similarity between user x and y, and r_{xi} and r_{yi} represent user x's and user y's rating of category i.

$$r_{xi} = \frac{\sum_{y \in N} s_{xy} \cdot r_{yi}}{\sum_{y \in N} s_{xy}}$$

Figure: Weighted Similarity Average

Triad Model. Now we go beyond just a positive and negative sign for the triad model from the preliminary analysis, edges between users and businesses can take on 5 values (1 to 5), and edges between businesses take the range -4, -3.5, -3.0...4.0. This will result in a greater number of triad configurations and better predictions. These new triads configurations break down the data to a finer granularity. We will analyze the frequency of such triads in the network and leverage that knowledge to make accurate predictions. We will first process a subset of the Yelp dataset to obtain the relative frequencies for each triad which we then use to make predictions. We define this subset of the dataset to be T which includes t number of random triads from the dataset; we define it like this to analyze how the performance changes as we vary the size of T. We then go through users who have made more than 1 review and look at each possible triad combination for the given businesses. So now, given a user u and business w in a given triad u, v, w. We know the edges of $u \rightarrow v$ and $v \rightarrow w$ and now want to predict the edge of $u \rightarrow w$. We do this by computing a weighted average of the rating and the relative frequency of that particular triad configuration. For example, if the triad u, v, w is closed with an edge from $u \rightarrow w$ of a rating of 4 25% of the time and a rating of 5 75% of the time, we would predict that edge to be 4.75. Now, we deal with the fact that a single edge can be in more than one triad. In these cases, we make the prediction to minimize the mean square error (MSE). To do this, we simply average the prediction ratings for each triad. For example, if an edge is in 2 dif-

ferent triads, one which gives a prediction of 3.5 and another of 4.25, we arrive at our final prediction of 3.875 for that edge.

VII. Results and Discussion

We will use mean square error (MSE) to evaluate the accuracy of our models. **Baseline**. First, we will define a baseline to compare against. The baseline will be linear regression with the following features: [user average stars, business review count, business stars]. We pull out 10000 random reviews from the dataset. Of those 10000, we use 70 percent for training and 30 percent for testing. The baseline obtains a MSE of 1.2983.

Collaborative filtering. From the results, we see that the weighted similarity prediction exhibited a lower MSE compared to the unweighted similarity prediction for a similarity group smaller than ten. this is not surprising as an unweighted prediction gives each member of the similarity group equal influence on the overall prediction, which may be detrimental if the discrepancy between the most and least similar group members is large. However, after the similarity group exceeds ten members, the difference between the performance of the two measures converge. We believe this is due to dilution in the similarity group weighting system. As more members are allowed into the group, the total sum of similarity measurements (which form the denominator of the weights) grows larger and thereby reduces the influence of the most similar members from the overall prediction.

Another trend that exists for both weighted and unweighted similarity prediction methods is that the MSE monotonically decreases when increasing the similarity group size. This implies that the more reviews that we consider when forming the overall prediction, even from users that may be not similar to the test user, the better the prediction. In the "All" case, when all users who have rated a particular restaurant are in the similarity group, we see that the lower limit of the MSE is a bit below 0.9 for both prediction methods. This trend suggests a tradeoff between computation complexity and prediction accuracy since a larger similarity group leads to more computation.

Lastly, we recognize that the current system does not predict low star ratings, those of one or two stars, very well. The reason for this is twofold. First, the distribution of aggregate star ratings is non-uniform as three, four, and five star ratings account for over 80% of the total ratings in our dataset. Second, the star predictions are calculated by summing parts of the actual ratings from members in a test users similarity group and therefore, any high rating of the restaurant for a review in that group will skew the overall prediction in that direction. However, even though this presents a problem for reducing MSE, it may not effect the overall results of this star prediction scheme. Even though predictions of one and two stars may be higher than their actual values, it is unlikely that they will be higher than the prediction of a three, four, or five star review.

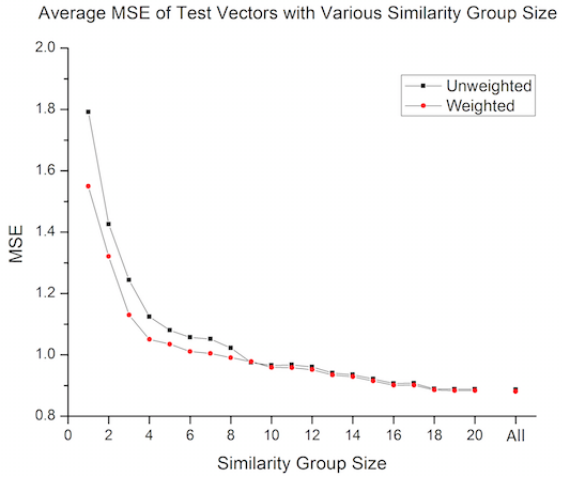


Figure: MSE vs Similarity Group Size

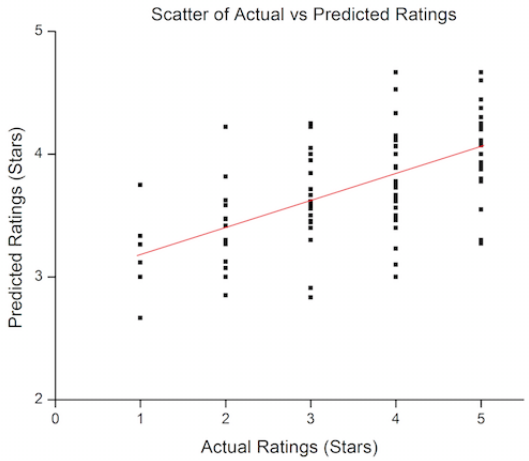


Figure: Predicted vs actual rating n= 20

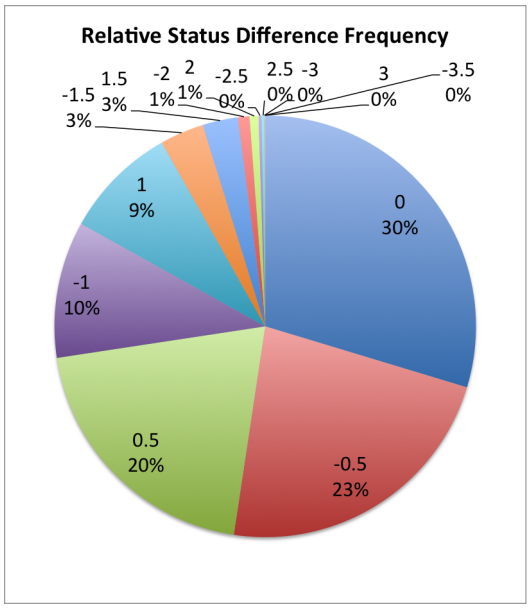


Figure: Relative triad frequencies of status differences between v and w, where a positive difference means v has a higher average rating than w. Note: percentages are rounded.

Triad Model. To test our model, we first pull out 500 random reviews of which the user has rated more than one business from the dataset, compute the relative frequencies of each triad configuration and do a prediction on each of the given reviews. We keep track of the actual and predicted ratings and the MSE for each prediction. We then compute an average MSE.

Given the computation resources needed to process these triads, we analyze the performance results for when we vary the number of triads ,t, we use to obtain the relative frequencies. When $t = 200$, the top 10 most common triad configurations and their relative frequencies are the following where (3,4,0.5) represents an edge of value 3 from $u \to v$, 4 from $u \to w$, and 0.5 from $v \to w$ (v has an average star rating 0.5 higher than w): (4, 4, 0.0): 0.052240527, (4, 4, -0.5): 0.035817683, (4, 4, 0.5): 0.033678358, (5, 5, 0.0): 0.032832876, (5, 4, 0.0): 0.032512618, (4, 5, 0.0): 0.032499808, (4, 5, -0.5): 0.031590274, (5, 4, 0.5): 0.026927314, (4, 3, 0.0): 0.023327611, (3, 4, -0.5): 0.023058594. From the figure we can see that the most frequent triads are those which match the status quo of the businesses most closely. This shows a relationship between the status of a given restaurant and the rating which a user u evaluates such a restaurant.

In the figure we have plotted MSE vs number of triads t. We see that when $t = 200$ is when the MSE plateaus and does not decrease as much. $t = 200$ is efficient with computation and performance. The MSE starts off at 1.214 for $t = 40$ and decreases to 0.379 for $t = 1000$. For the most balanced case of $t = 200$ the MSE is 0.42. From $t = 200$ on, we get diminishing returns for the additional computational resources required.

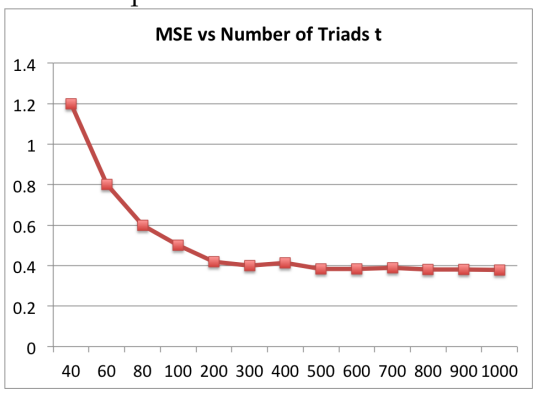


Figure: MSE vs number of triads t processed to obtain the relative frequencies of triads.

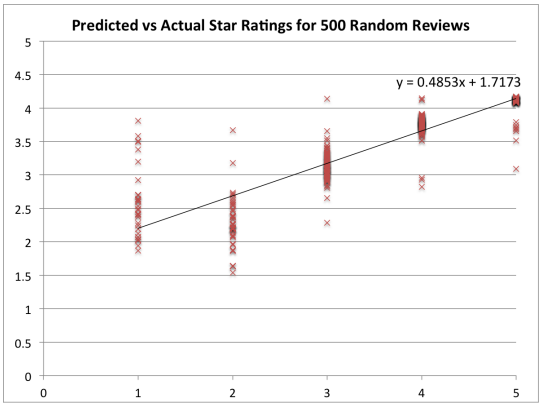


Figure: Predicted vs actual rating for 500 random reviews for t = 200

VIII. Challenges

Difficulties we ran into mainly dealt with the large dataset size. To avoid reading and parsing the data every time we ran our program, we utilized mongoDB. Through Mongo, we are able to store our dataset on a server and retrieve specified users and businesses according to their id. Although the time to set this up is significant, the benefits made it well worth it. In addition, running tests on such large datasets is time consuming. To alleviate this issue, we parallelized the tests into different processes through python.

IX. Future Work

After implementing and testing the current Yelp rating prediction system, we recognize several areas of improvement for future work.

Enhanced Low Star Predictions As mentioned previously, this system does not handle the prediction of low star ratings well due to the non-uniformity of the distribution of one and two star ratings in the aggregate dataset. We believe it may be interesting to consider the businesses aggregate rating distribution and incorporate a random weighted variable from that distribution to the prediction.

Improvement of Similarity Measure Weighting. A key part of the collaborative filtering Yelp prediction system is the similarity algorithm used to measure how similar two users are in restaurant tastes. In the current implementation, the magnitude of the similarity measure between the test user and a member in the similarity group is used to assign a weight to the similarity group members rating for the restaurant in question. Thus, the weighting mechanism used is linear in the similarity measure. However, we suspect that this is non-optimal, since as the similarity group size increases, the weighted and unweighted similarity predictions converge due to a larger total weight denominator. Thus, we believe using a non-linear weight distribution model will better account for star rating predictions in large similarity groups.

Extension to Recommendation Engine Finally, an application of our rating prediction model is as a part of an overall recommendation engine. The most simplistic way of implementing such a recommendation engine would be to rank all restaurants by predicted star rating for a given user and serve the recommendations in that order. More advanced algorithms can also incorporate location-aware services, sentiment analysis on what a user is currently craving, and social hot-spots. This recommendation engine would therefore attempt to curate a personalized dining experience for each user.

X. Conclusions

We have shown the importance of the status of a business on the ratings which it receives. When a user evaluates a business, the status of the business carries weight. With the end goal of predicting Yelp star ratings, we have analyzed two main models: collaborative filtering and the triad model. Even though collaborative filtering is a more complex and computationally expensive model, it is outperformed by the triad model by over two times. Collaborative filtering achieves an MSE of approximately 0.9 for n=20. The triad model achieves an MSE of approximately 0.4 for t=200. The triad model exemplifies the efficiency and simplic-

ity of leveraging the dataset and the network structure to make predictions. It is a simple model which is able to capture additional information within a simple triad configuration and then utilizes the relative frequencies of different triads. The triad model shows the power of analyzing a dataset from a network perspective and the additional insights which that brings.

References

[1] Ashton Anderson, Daniel P. Huttenlocher, Jon M. Kleinberg, Jure Leskovec: Effects of user similarity in social media. In *Proc. WSDM*, 2012.

[2] J. Leskovec, D. Huttenlocher, and J. Kleinberg. Governance in social media: A case study of the Wikipedia promotion process. In *Proc. ICWSM*, 2010.