METHOD

The proposed methods begin in community detection, user influence definition and visualization in different communities. Based on these two results, we provide recommendations for users through combining the information of ratings and users' influence. An interactive recommendation and data visualization interface is also provided.

COMMUNITY DETECTION

The first step of our community detection process is to build a social graph based on restaurants preference of users. So the nodes in our social graph is 13969 users, and we use the Jaccard Index for building edges in our graph. The basic Jaccard Index is defined as following:

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}$$

Based on this basic definition, we added another similarity factor of users' rating scores. And the refined Jaccard Index is:

$$J'(A,B) = J(A,B) + \frac{\sum |R(A) - R(B)\,|}{10 * N(A \cap B)}$$

In this equation, R(A) and R(B) means the rating stars of user A and B for a certain restaurant. N(A∩B) means the number of common restaurant reviews for user A and B. After trying different cutoff values for Jaccard Index and considering the size of our graph, we chose to use 0.4 as the threshold to build our edges. So as the result, our social graph contains 272589 edges.

The second step of our community detection is to use Greedy algorithm to detect communities, and this algorithm needs first define a modularity index Q of within-community edges as follows:

$$Q = \frac{1}{2m} \sum_{vw} \left[A_{vw} - \frac{k_v k_w}{2m} \right] \delta(C_v, C_w)$$

After writing $\delta(C_v, C_w) = \sum_i \delta(C_v, i)\, \delta(C_w, i)$ and start off with each vertex being the sole member of a community of one, we define our similarity matrix as follows:

$$\Delta Q_{ij} = \frac{1}{2m} - \frac{k_i k_j}{(2m)^2}$$

And then the Greedy iteration steps are the following:

1. Calculate the initial values of ΔQij and ai according to (8) and (9), and populate the max-heap with the largest element of each row of the matrix ΔQ.

2. Select the largest ΔQij from H, join the corresponding communities, update the matrix ΔQ, the heap H and ai (as described below) and increment Q by ΔQij .

3. Repeat step 2 until only one community remains.

USERS' INFLUENCE DEFINITION

In the above section of recommendation system, just the similarity between different users' pairs are considered. As we have clustered the users into different community in the previous section, we can provide recommendation to users based on other users' choices inside the same community.

However, different users may hold different influence on the other users inside the community. Influence is a measure of the effect of a user on the recommendations from a recommender system. Here we consider the users' influence and use this criteria to improve the performance of our proposed system. The measure of the users' influence can be used into the ratings-based recommender system.

There are two ways to define the users influence in the communities. We can use some characteristics in Graph to describe the users influence. PageRank works by counting the number and quality of links to a page to determine a rough estimate of how important the website is. The underlying assumption is that more important websites are likely to receive more links from other websites. In our case, we think the community contains users with some common appetites, but are not that similar, as you can see with a community there are also many small tight communities. And this is similar to the website cases, we can treat any user as important one if he connects two small tight communities within the big community and with some good attributes like the amount of view. Thus the user with higher page rank score is important and considered as the influence definition in that large community.

The model can be obtained as follows:

$$PR_i(t) = (1-c)\sum_{j=1}^{n} a_{ji}\frac{PR_j(t-1)}{k_j^{out}} + \frac{c}{n}$$

In order to consider the users' influence, another way to describe the users influence is to exclude a user and use some metrics to measure the change in predictions caused by this removal. Suppose that S is the set of the users in a particular community, and the recommendation model with this set is denoted asM_S, and the model after removing user 'i' is denoted as M_{S-u_i}, then here define the influence factor IF_{u_i} as the number of times when the following expression happens:

$$|P_{ja}(M_S) - P_{ja}(M_{S-u_i})| \geq \theta \quad \forall j \neq i$$

Where $P_{ja}(M_S)$ is the prediction of user j on the item 'a' with model M_S and $P_{ja}(M_{S-u_i})$ is the prediction of user j on the item 'a' with modelM_{S-u_i}. θ is defined here as the threshold used to identify the prediction change. That is to say, small value means a more sensitive to the model change. However, due to the tremendous computation in this method, we only consider the first way to define the users influence.

RECOMMENDATION SYSTEM DESIGN

In the clustering process, we filtered all the data from Pittsburgh and clustered the user into several communities. We develop our recommendation system based on these communities, namely, we compute the pair-wise distance for a certain user only in the community where the user belongs to.

Here the collaborative filtering (CF) recommender systems are first considered in recommendation in different communities. In this approach, the users' preference data is represented in a user-item matrix which have m users and n items, and the (i, j) element of the matrix is the rating for the user i on item j. In the collaborative filtering method, similarities between different users' pair 'u' and 'v' are computed by the equation below:

$$S(u,v) = \frac{1}{1 + D(u,v)}$$

where D is the distance between the pair u and v, which is defined as

$$D(u,v) = \sqrt{\sum_{k=1}^{n} (u_i - v_i)^2}$$

The weight here is used to identify the similarity between different users' pairs, which is used in the next step to predict and recommender the item to the user. In order to achieve recommendation, during the prediction step, Prediction on item 'a' for user 'i' is computer by picking the most k similar users who have rated item a, and use the following equation to apply a weighted average of deviations from the selected users' means.

$$P_{ia} = \bar{R}_i + \frac{\sum_{u=1}^{k} (R_{ua} - \bar{R}_u) W_{iu}}{\sum_{u=1}^{k} W_{iu}}$$

Consider the dataset in Yelp. From the filtered Pittsburgh data, we collect "user_id" (the id of user), "business_id" (the id of the restaurant) and "stars" (the stars given by the user to the restaurant, range 1 to 5), to establish a sparse matrix M, which size is (number of user)*(number of restaurant). In $M_{3,4}$, for example, it may contains the rating information of number 4 restaurant rated by user number 3. From this matrix, we compute pair-wise distance, only if when two users have at least one restaurant they both rated, and they are in the same cluster.

Consider the users influence IF which has been calculated in the previous section, then we can give the following updated prediction method as follows

$$P_{ia} = \bar{R_i} + \frac{\sum_{u=1}^{k}(R_{ua} - \bar{R_u})W_{iu}IF_{u_i}}{\sum_{u=1}^{k} W_{iu}IF_{u_i}}$$

Through using the above equation, we can predict the users' choice and make the personal recommendation to the user in a more proper way. This model not only consider the ratings but also conclude the difference among the users in the community. From our method, we can see that the neighbors with higher similarity and larger influence could have a larger weight for predicting the predicted rating for a specific user.

Appendix: REFERENCES

[1] Resnick, Paul, et al. "GroupLens: an open architecture for collaborative filtering of net news." Proceedings of the 1994 ACM conference on Computer supported cooperative work. ACM, 1994.
[2] Guo, Guibing. "Improving the performance of recommender systems by alleviating the data sparsity and cold start problems." Proceedings of the Twenty-Third international joint conference on Artificial Intelligence. AAAI Press, 2013.
[3] Sarwat, Mohamed, et al. "LARS*: An efficient and scalable location-aware recommender system." IEEE Transactions on Knowledge and Data Engineering, 26.6 (2014): 1384-1399.
[4] Davoodi, Elnaz, Keivan Kianmehr, and Mohsen Afsharchi. "A semantic social network-based expert recommender system." Applied intelligence 39.1 (2013): 1-13.
[5] Liu, Xin, and Karl Aberer. "SoCo: a social network aided context-aware recommender system." Proceedings of the 22nd international conference on World Wide Web. ACM, 2013.
[6] Bostandjiev, Svetlin, John O'Donovan, and Tobias Höllerer. "Tasteweights: a visual interactive hybrid recommender system." Proceedings of the sixth ACM conference on Recommender systems. ACM, 2012.
[7] Pons, Pascal, and Matthieu Latapy. "Computing communities in large networks using random walks." Computer and Information Sciences-ISCIS 2005. Springer, 2005. 284-293.
[8] van Gennip, Yves, et al. "Community detection using spectral clustering on sparse geosocial data." SIAM Journal on Applied Mathematics 73.1 (2013): 67-83.
[9] Xie, Jierui, Stephen Kelley, and Boleslaw K. Szymanski. "Overlapping community detection in networks: The state-of-the-art and comparative study." ACM Computing Surveys, 45.4 (2013): 43.
[10] Yang, Jaewon, Julian McAuley, and Jure Leskovec. "Community detection in networks with node attributes." Proceedings of the 13th international conference on Data Mining. IEEE, 2013.
[11] Fortunato, Santo. "Community detection in graphs." Physics Reports 486.3 (2010): 75-174.
[12] Michael Luca, Georgics Zervas, Fake It Till You Make It: Reputation, Competition, and Yelp Review Fraud. May, 2015.
[13] Michael Blanding, The Yelp Factor: Are Consumer Reviews Good for Business? October, 2011.
[14] Young-shin Lim, Evaluating the Wisdom of Strangers: The Perceived Credibility of Online Consumer Reviews on Yelp, 2015.