# Yelp Recommendation System via User's Personality and Sentiment Analysis in Reviews

Ruyin Shao
University of Southern California
ruyinsha@usc.edu

Di Huang
University of Southern California
huangdi@usc.edu

Kai Wang
University of Southern California
wang993@usc.edu

Xuezheng Tao
University of Southern California
xuezhent@usc.edu

## ABSTRACT

Recommending restaurants that fit users' interest can improve user experiences of online review platforms like Yelp and bring more customers to restaurants. In this paper we build a recommendation system that combines user-based, restaurant-based collaborative filtering and personalized recommendation based on sentiment analysis on review text. We extract related features of restaurants from Yelp reviews and use k-NN to find a candidate set of similar users for each given user based on their preferences. Then we calculate restaurant similarities based on rating in reviews. For personalized recommendation, we extract 7 features from the review text and transform features into vectors, which we used to build user and item profiles and calculate cosine similarities as personalized scores. We then use linear regression to set different weights for the personalized preference scores and collaborative filtering results. We use the weighted prediction to be our final recommendation. Our experiments show good precision and recall values produced by our system.

## KEYWORDS

k-NN, collaborative filtering, personalized recommendation

## 1 Introduction

Customers have different preferences when they are selecting restaurants. Some users may prefer quiet environments while others may think that the taste is more important. Traditional recommendation systems that recommend only based on rating data cannot fully meet different users' needs.

To address this challenge, some researchers explore the topics mentioned in review text with Latent Dirichlet Allocation（LDA) for rating predictions [1], which is proved to be helpful. We attempt to combine advantages of collaborative filtering and the strength of review text in reflecting users' preferences.

In addition to traditional collaborative filtering, we extract related features like "flavor" and "funny" in user and restaurant data of Yelp dataset and transfer them into vector representations. These vectors will be used to calculate Euclidian distances between users for finding similar users with k-NN algorithm. Selecting similar users based on k-NN can narrow the range of candidates to a great extent. Also, it can make predictions of collaborative filtering more precise since traditional user-based model computes similarities based only on set of items that both users rated.

Review text also contains valuable information that can make recommendation more precisely. It contains user's attitudes towards some aspects of restaurants such as environment, parking, taste and price. We extract related keywords and transfer the review text into vectors based on sentiment analyzing results. We use the cosine similarity calculated based on these vectors to improve the precision of prediction.

## 2 Dataset

Our original dataset comes from Yelp Dataset Challenge. We select user, business, and review data from the original set and filter out the inactive user to perform user similarity calculation. After filtering out the user who has less than 10 friends and 10 reviewed restaurants, there remain nearly 64K users, 2 million reviews written by these users and 150K restaurants.

We also separated out three test sets that include 1000, 3000, and 100,000 reviews.

## 3 Methods for Our Recommendation System

We combine several algorithms that help determine user's focus and identify similar restaurants that the user prefers to make our final recommendation. We set different weights for different algorithms in rating prediction. For each user, we find his similar neighbors with k-Nearest Neighbor(k-NN) algorithm. We then calculate user similarities and item similarities and use them for item-based and user-based collaborative filtering to predict the ratings of users for the candidate restaurants. Besides, we extract personalized information from users' review text, conduct sentiment analysis for the text and transform them into vector representations to make our recommendation meet each user's special preferences. Our final prediction will be calculated based on the weighted sum of the analytical result from collaborative filtering and sentiment analysis results. The flow chart of our algorithms is shown in figure 3.1.

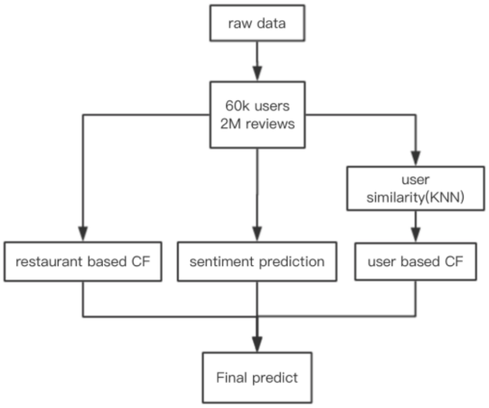

**Figure 3.1 Flow Chart of Our Weighted Algorithm**

### 3.1 User Similarity in Personal Interest

We find that active users who have more than 10 reviewed restaurants might choose to visit restaurant based on his own preference rather than the influence from friends. Previous work mainly focuses on extracting the common interest among users and their friends. [2-3] Instead of focusing on merely the user's social network, we look for user's neighbors who have similar personal interest. We add these neighbors to user's social network group to form a community that indicate user's interest based on their friends and their similar neighbors.

For a given user, we use k-NN algorithm to narrow the range of similar users by selecting users that have the similar preferences with the given user. We represent each user as a sparse vector like {"Mexican": 0.6, "Chinese": 0.22, "Funny": 0.3 ……}, where each label is extracted from the original dataset and the value is the frequency of the user going to that kind of restaurant, namely, how many times the user goes to a certain type of restaurant, divided by the total number of review records the user has left in Yelp.

The k-NN algorithm can pick the top $n$ users that have the smallest Euclidian distances with the given user from candidate data sets. We choose to extract $n$=10 similar users for each given user.

### 3.2 Personalized Recommendation Based on Sentiment Analysis of Review Text

As some users may care more about a restaurant's food while others may be more concerned with the service, we extract the related features from users' review text to make personalized recommendation. Specifically, we transform each piece of review text into a 7-dimensional vector representation. For example, $Review(uid, bid)$= [0, 0, 0.61, -0.1, 0.87, 0, 0.34] denotes a piece of review made by a user $uid$ and the restaurant $bid$. Each dimension denotes the corresponding value of a feature among "food", "service", "parking", "noise", "crowded"(too many people), "price" and "environment". We extract different parts of the review text that mention different features and use sentiment analyzing functions in NLTK library to calculate scores for each dimension in the vector. The sentiment score is in the range of [-1, 1], depending on whether the user's attitude is positive or negative and how strong it is. If a feature is not mentioned in the review, we set it as default score 0.

After transforming all the review text into vectors, we build restaurant profiles by summing up the vectors of corresponding restaurants as following:

$$V_{bid} = \sum_{uid} Review(uid, bid) \qquad (1)$$

Similarly, we compute the user profiles by summing up the vectors of corresponding users after taking absolute values as formula (2) shows, where $|Review(uid, bid)|$ means taking absolute value for each element in the vector $Review(uid, bid)$.

$$V_{uid} = \sum_{bid} |Review(uid, bid)| \tag{2}$$

Finally, we calculate the cosine similarity between each user and each restaurant and conduct personalized recommendation based on these results. This is expressed as:

$$CosSimilarity(uid, bid) = \frac{V_{uid} \bullet V_{bid}}{\|V_{uid}\| \, \|V_{bid}\|} \tag{3}$$

### 3.3 Final Recommendation

We combine item-based and user-based collaborative filtering to predict ratings of a user for a given restaurant. Item-based collaborative filtering will consider the restaurants the user has rated and predict the rating according to scores of these similar restaurants, as formula (4) shows. We use $rated(x)$ to denote the set of restaurants that user $x$ has rated, and in order to reduce time complexity, we only select 20 restaurants by random sampling when calculating the similarity value. We use $sim(x, y)$ to denote the cosine similarity of restaurant $m$ and y , and $r(x, m)$ to denote the rating score of user $x$ for restaurant $m$.

$$IP_{x,y} = \frac{\sum_{m \in rated(x)} sim(m,y) r_{x,m}}{\sum_{m \in rated(x)} sim(m,y)} \tag{4}$$

User-based collaborative filtering considers the similar users' ratings as formula (5) shows, where $N(x)$ is the set of similar users we get by k-NN algorithm. $sim(x, b)$ is the similarity value we get by transforming the Euclidian distances between users into the range of (0, 1].

$$UP_{x,y} = \frac{\sum_{b \in N(x)} sim(x,b) r_{b,y}}{\sum_{b \in N(x)} sim(x,b)} \tag{5}$$

And we adjust the predicted score by setting some weights for the personalized preference score, as (6) shows. Here $Psim(x, y)$ denotes the cosine similarity. We adjust the weight values of $w_1$, $w_2$, $w_3$ and *bias* through linear regression in the training data. Linear regression is a machine learning approach to build the linear relationship between dependent variables and several dimensions of independent variables [4]. Our experiments show that when we set $w_1$, $w_2$, $w_3$ and *bias* to be 1.0, 0.3, 0.1 and -1.5, it will produce satisfying results. Finally, if the prediction score value of user x for restaurant y is no less than 4.0, then we will recommend this restaurant to the user; otherwise we will not recommend it.

$$Pred_{x,y} = w_1 IP_{x,y} + w_2 UP_{x,y} + w_3 Psim(x, y) + bias \tag{6}$$

## 4 Results

### 4.1 User Profiles Generated By k-NN and Sentiment Analysis

By applying k-NN to identify user's neighbors and concatenate the interest with user's social network, we generate a sample user profile based on the frequency of the restaurant attributes they are looking for.

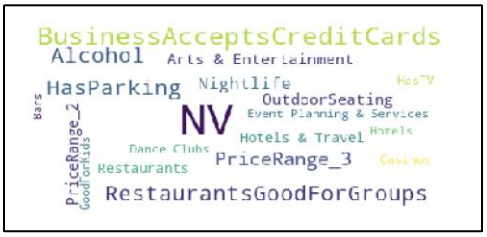

**Figure 4.1.1 User Profile of Personality based on k-NN results**

After sentiment analysis of related features from review text, we calculate the average values of user profile vectors in each dimension and represent it as a bar graph, shown in Figure 4.1.2. It shows that most users care most about service and food of restaurants.

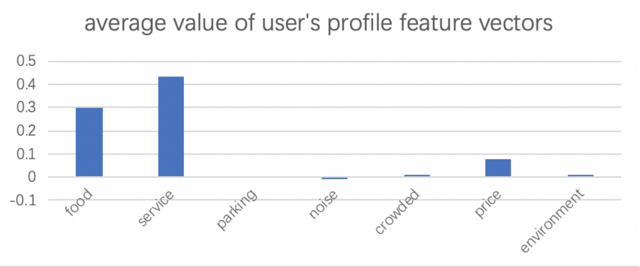

**Figure 4.1.2 Sample User Profile Based on Sentiment Analysis of Review Text**

## 4.2 Evaluation

We use 3 indicators to measure the quality of our predictions. They are accuracy, precision and recall. *TP*, *TN*, *FP*, *FN* denote the counts of true positive, true negative, false positive and false negative value. Accuracy indicates the proportion of the cases where our algorithm predicts the correct decision. Precision indicates the correct recommendation ratio in all predicted recommendations. Recall indicates the correct recommendation ratio in all possible recommendation.

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \tag{7}$$

$$Precision = \frac{TP}{TP+FP} \tag{8}$$

$$Recall = \frac{TP}{TP+FN} \tag{9}$$

The results of running user based collaborative filtering, item based collaborative filtering and our weighted algorithm are listed below:

**Table 4.1. Comparison of traditional collaborative filtering results and our weighted algorithm result**

| Algorithm | Accuracy | Precision | Recall |
|---|---|---|---|
| user based CF | 61.49% | 79.69% | 54.74% |
| item based CF | 67.41% | 92.33% | 54.58% |
| our algorithm | 69.95% | 93.29% | 58.83% |

As we can see from tables and figures, all algorithms have a good precision rate which means most recommended restaurants found by algorithms do have a good rating and they might be good choices for users. However, the recall rates are not ideal, which means the algorithms are not good at acquire customers' subliminal wanted restaurants. By comparing these three algorithms, we can conclude that our algorithm has better performance than the others.

## 5 Future Work

We need to lift the restriction on user friend number and apply the k-NN algorithm to all users who have enough reviews in order to get better predictions. Also, want to represent category, price and geographical location as 3 different features, and we can use logistic regression or other ways to set appropriate weights for them.

Besides, instead of setting the recommendation baselines as 4.0, we will set different baselines for different users based on their average ratings which can serve as a bias term.

## 6 Conclusion

In this paper, we combine several algorithms to form a personalized recommendation system. We use k-NN to find similar users of a given user and transform the Euclidian distance into a similarity measure. We then use these results to implement user-based collaborative filtering and use cosine similarity between restaurants for item-based collaborative filtering. To take user's personal preferences into account, we extract several features from the review text and transform reviews into vectors based on sentiment analyzing results. We combine the three recommendation results by setting different weights for them according to linear regression and our experiments show that it can provide accurate recommendation for users. We also extract more information from user profiles and figure out the related features that each user may be more concerned about, like the food or the service, and can present the analyzing results of related features to users when we recommend the restaurants to them.

## REFERENCES

[1] Linshi J.Personalizing Yelp star ratings: A semantic topic modeling approach[J]. Yale University, 2014.

[2] X. Yang, et al, "Circle-based recommendation in online social networks," in KDD'12

[3] M. Jiang, P. Cui, et al, "Social contextual recommendation," in CIKM'12, Oct.2012.

[4] Yan X, Su X. Linear regression analysis: theory and computing[M]. World Scientific, 2009.

## APPENDIX

[1] Github link:
https://github.com/qianlizimu/inf553Project