**Hybrid Recommender Systems: Summative Assignment**

# INTRODUCTION

## Yelp has quickly emerged as the most popular review site which has resulted in Yelp receiving a massive amount of user and business data. The data varies from people’s preferences and personalities to reviews, ratings, and general information provided by the community about any business. Today, consumers around the world can choose from even the simplest services. Although all this information is easy to obtain, it is often difficult for people to make these choices based on the raw data provided by yelp. A recommendation system that uses their personal preferences and the preferences of similar users to suggest the potential best choice for them makes it easier for users to solve this problem.

## *A. Problem Statement*

Yelp online reviews are a very valuable source of information, and users can choose where to visit or what to eat from among the many available options. But due to a large number of comments, it is almost impossible for users to browse all the comments and find the information they want. To solve this problem, we created a recommendation system. Recommendation systems usually generate recommendation lists in one of two ways: collaborative filtering or content-based filtering. The collaborative filtering method builds a model based on the user’s past behavior (previously purchased or selected items, and/or the numerical ratings of these items) and similar decisions made by other users; then uses the model to predict the items (or items) that may be of interest to the user Rating).

In this project, we aim to predict the rating of

restaurants listed in the Yelp dataset based on the reviews given by the users. And then recommending restaurants to the users using a hybrid recommender system.

# METHODS

## *A. Dataset*

The dataset used for this task was obtained from the Yelp dataset challenge [1], which consists of 1.6M reviews and 500K tips by 366K users for 61K businesses. The dataset consists of five files - business, comments, users, check-in, and prompts. Operational and review documents are mainly used for this predictive task. More specifically, it includes 61184 businesses, 1569264 comments, and 366715 user data. All data are in JSON format.

We converted the data in JSON format to CSV files and the two main CSV files are business.csv and reviews.csv. The reviews after 2019-10-1 are selected as the data set.

## *B. Feature Selection*

The structure of the rating data is as follows: (user ID, item ID, rating), where the user ID and item ID are unique integer values, and the rating range is 1-5. We also mark the rating>4.0 as like to evaluate our recommendation system.

For the convid19\_business data set, since it is not in the same vector space as the previous data set, we will use it as an additional feature for prediction without participating in the training process.

## *C. Collaborative filtering*

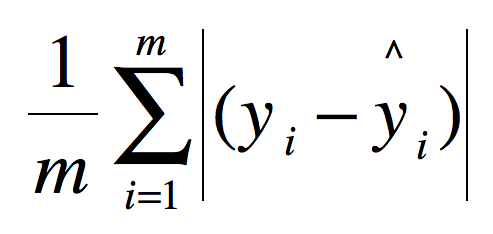
The idea of collaborative filtering is to find users in communities that share appreciation. If two users share the same or almost the same rated items, their tastes are similar. Such users establish a group or so-called neighbor. Users will receive suggestions on projects that have not been rated before but have received positive comments from community users.

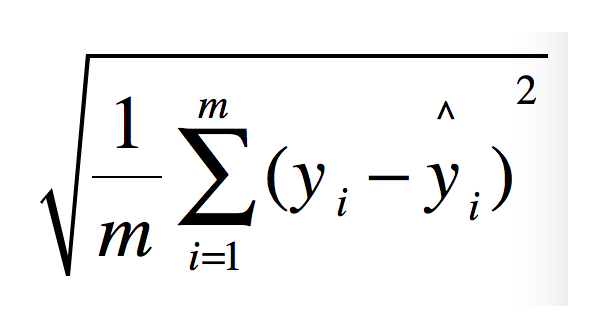
*D. Content-based filtering*

The content-based recommendation engine can be used with existing user profiles. Personal data contains information about users and their tastes. Taste depends on the user rating of different items. Usually, every time a user creates a profile, the recommendation engine will conduct a user survey to obtain the initial information about the user, to avoid new user problems. In the recommendation process, the engine will compare the products that users have actively evaluated with those that users have not yet evaluated, and look for similarities. Users will be recommended projects similar to positive reviews. Here, according to the taste and behavior of users, we can build a content-based model by recommending articles related to user taste.

*F. Evaluation Methods*

The evaluation was done using an ofﬂine experiment speciﬁcally on user’s ratings. The accuracy of rating predictions was evaluated using Mean Absolute Error (MAE)and RMSE（Root Mean Squared Error）, which is supplied by the following equation [2]:





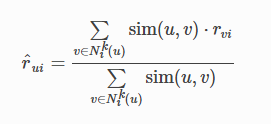
Another way to evaluate is to use a confusion matrix to calculate precision and recall,

Precision and recall can be calculated by the following formula [3],

# IMPLEMENTATION & Evaluation

The k-NN inspired algorithm [4] is our baseline recommender, which is also a basic collaborative filtering algorithm.The prediction is set as:



Another one is SVD, a matrix factorization-based algorithm.The results of KNN-based Recommender:

RMSE: 1.495

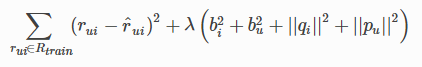
MAE: 1.268

The famous SVD algorithm, as popularized by Simon Funk during the Netflix Prize. When baselines are not used, this is equivalent to Probabilistic Matrix Factorization.

The prediction is set as:



If user  is unknown, then the bias  and the factors are assumed to be zero. The same applies to item  with  and .

To estimate all the unknown, we minimize the following regularized squared error:

The minimization is performed by a very straightforward stochastic gradient descent:

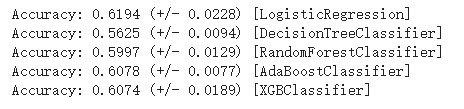
The results of SVD Recommender:

RMSE: 1.394

MAE: 1.138

The third method is to feed the features of the item and the user, use the model to score the item, and finally select the top-k recommended items according to the score.

Through cross-validation, after feature engineering, we compared five scoring models and gave the test results.



*E. Hybrid method*

We use a linear weighted summation method to combine the two personalized recommendation models, where the weight is determined based on the results of the two models tested separately, the predicted formula is as,

*F. Ethical issues*

The first ethical issue is whether the user has been informed in advance of the possible future uses of the data when collecting the data.

The second issue is whether it is visible to the company when the data is used, such as private data such as personal address.

The third issue is whether the recommendation system will be abused, such as raising the price of items that users may prefer.

We can avoid leakage of user information through edge computing.

# Conclusion

In conclusion, we have experimented with various feature selection techniques and various supervised learning algorithms to predict star ratings of the Yelp dataset using review text alone. We evaluated the effectiveness of different algorithms based on MAE and RMSE measures.

We extended our model to incorporate the rating of restaurants predicted as features to perform user restaurant recommendations by clustering. We implemented content-based and collaborative filtering. Both the models work well. But we realized that content-based leads to over-specialization i.e., recommended restaurant is like already visited/reviewed restaurant and may not be useful for the user. Whereas collaborative filtering relies on past preferences or rating correlation between users. However, this technique can lead to bad predictions if the restaurant is unpopular and very few users have reviewed that restaurant. So, a hybrid model that considers the aspects of both content-based and collaborative filtering should give the best results.

Reference

[1] Yelp Challenge Presentation: https://www.yelp.com/dataset/download

[2] https://en.wikipedia.org/wiki/Mean\_squared\_error

[3] https://en.wikipedia.org/wiki/Precision\_and\_recall

[4] https://surprise.readthedocs.io/en/stable/index.html

[5] Yin Y, Chen L, Xu Y, et al. QoS prediction for service recommendation with deep feature learning in edge computing environment[J]. Mobile Networks and Applications, 2019: 1-11.