Yelp Restaurant Recommender System

Natalie Kim

Problem Statement & Data Introduction

Provide users with restaurant recommendations based upon either previous reviews made or the attributes of the restaurants. The system aims to enhance user satisfaction by suggesting relevant dining options and improving user engagement on the Yelp platform.

Initial 5 data sets provided as json objects: business, users, reviews, tips, and check-in

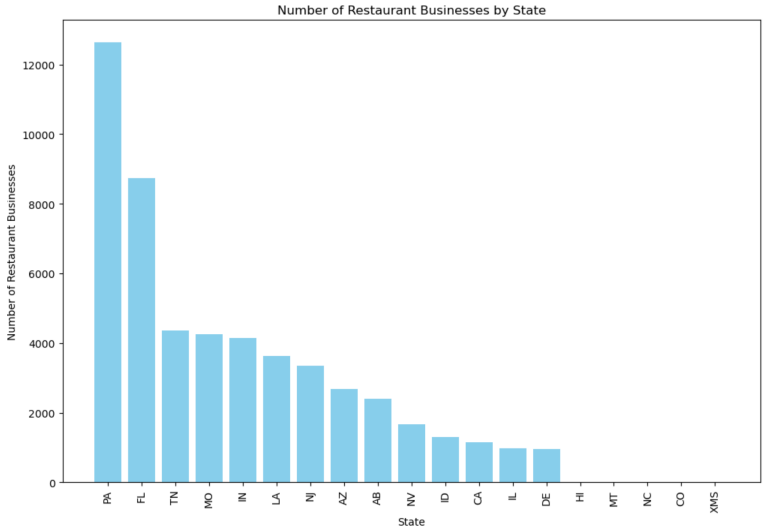
Assumptions/Hypotheses

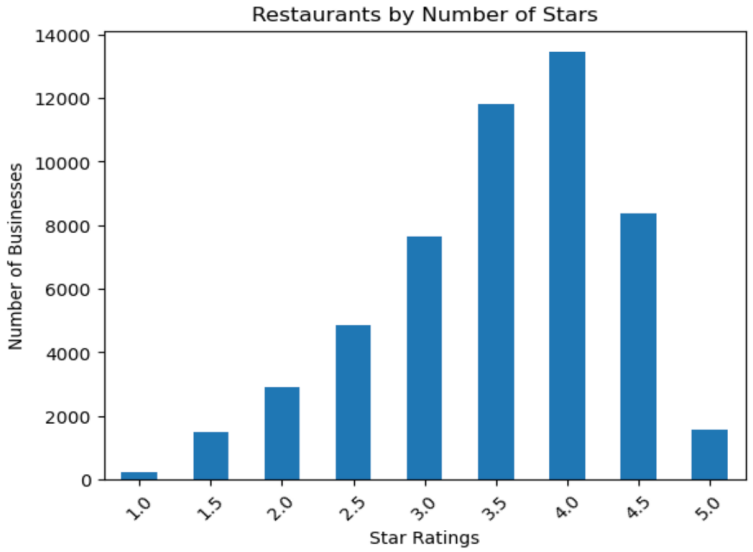
- If a user is interested in restaurants, they will not be looking for other types of businesses. Therefore, we will be training on restaurants
- User looking for restaurants in a different state. This limits our data to Pennsylvania.
- For simplicity and due to the size of the data, I limited this project to only the provided input star scores to evaluate metrics.
- According to Yelp website: Only the reviews that Yelp recommended at the time of data collection are included.
- Users rate restaurants based on their experiences, and these ratings reflect their preferences.
- The user-item interactions (ratings) can be used to infer user preferences and restaurant characteristics.

https://www.yelp.com/dataset/documentation/faq

Exploratory Data Analysis

- Over 12,000 restaurants in Pennsylvania. If we dig deeper, then we see that Philadelphia is the city with the greatest number of restaurants: 5854
- Scoring is number of stars ranging from 1 to 5
- There are many categories and subcategories for each restaurant
- Data sets: users, tips, and check-ins found not to be useful for this project





Feature Engineering & Transformations

- Split the data into 90:10 train split due to the smaller amount of useable data
- Then created
  - user-item matrices for collaborative-filtering, and
  - dummy matrices for attributes and categories for content-based recommendations by flattening out these columns
- Create similarity matrix using cosine similarity the user-item matrices and the dummy matrices above.
- Additionally normalized the user-item matrices using the mean and ensured columns matched between both the training and test sets.

Two Proposed Approaches

- Content Based Filtering using K Nearest-Neighbors
  - Recommend restaurants based upon the business' qualities
  - Using GridSearch Cross Validation, found that the optimal k was 30.
  - Then used the training set of the similarity matrix to train the model
  - Not limited by cold-start issue presented by collaborative based filtering
- Collaborative Based Filtering using SVD
  - Recommend restaurants based on the past ratings and preferences of other users.
  - Can exploit the collective wisdom of all of Yelp's users and is not burdened by all of the different attributes of restaurants

Proposed Solution

- Based upon both business reasoning as well as model evaluation metrics, I determined the best model to be the collaborative based filtering model.
  - Although the content-based filtering using KNN with number of neighbors = 30 performed mildly, once I applied regularization, the RMSE significantly dropped and improved the model.
  - I applied L2 Ridge Regularization primarily because of my choice to use SVD (i.e., matrix factorization) for the collaborative model and because I knew the user-restaurant matrix would be sparse since many users do not frequently write reviews for many restaurants.
- The content and collaborative models were evaluated based upon accuracy and RMSE. The RMSE of the content-based model did not match up to the collaborative model.

Results/Learnings from Methodology

- Collaborative Filtering:
  - Achieved an RMSE of 3.926 on the test set, indicating good predictive performance.
- Content-Based Filtering
  - Although the model did not display any signs of overfitting due to its consistent accuracy across train and test sets. The lower accuracy did not prove to be enough when compared to the collaborative filtering results.
  - Achieved a test accuracy of 51.54% using KNN with the optimal number of neighbors.
- Learnings
  - Collaborative filtering using SVD with regularization performed better than content-based filtering for this dataset
  - Regularization helped prevent overfitting and improved generalization

Learnings/Future Work

- The content-based model shows promise in the lack of overfitting, however, it can be improved. Some ideas on ways to improve include:
  - Tuning the hyperparameters further
  - Adjusting the calculation of the ratings to create a super score that is a combination of multiple values rather than use only the star score
- This project challenged me to be more proactive in looking out for overfitting. Though requiring some further work on the data, the regularization improved my RMSE and paid off in the end.
- Because the models are still either collaborative or content, they both have their limitations. If I were to have more time and plan more for processing power, I would use a combination of the two in order to provide a much better recommendation.
  - This is based on both analytics and business intuition as providing only user input OR restaurant attributes is not enough

Future Work

- Seek out combining collaborative and content-based filtering to leverage the strengths of both approaches
- Explore keras neural network options for deep-learning options to improve upon the recommender systems.
- Revise code to be more scalable. I ran into many issues due to the large size of the ‘reviews’ dataset
- I would enjoy trying to expand this project to either other cities or other business and challenging the model to classify between further variations of the data.

Resources

- Code: https://github.com/nskim1/ADSP_ML_FinalProject.git
- https://www.kaggle.com/datasets/yelp-dataset/yelp-dataset