410 Team Project Final Report

Team Name: DuoDuo

Project Topic: Restaurant Concierge

Team Members:

Rhea Chen	xinyuc11	xinyuc11@illinois.edu (captain)
Jingjing Yao	jyao27	jyao27@illinois.edu
Xian Chen	xianc2	xianc2@illinois.edu

1. Overview

1.1 Introduction

This project aims to develop a simple graphical user interface applicationuser-friendly and efficient restaurant concierge service for tourists with limited time in a city. The application will address the common pain point of finding the best restaurants during short trips by simplifying the search process and providing curated recommendations based on user preferences.

1.2 Objectives

The existing restaurant apps offer filtering options, they often fall short in:

- **Personalization:** Most apps only allow users to apply one filter at a time
- **Objectivity:** Most platforms prioritize advertising over user experience, displaying sponsored restaurants higher in search results.

We utilize the BM25 algorithm, which ranks restaurants based on reviews that match some positive words. This ensures that the top results are consistently good and meet the user's specific needs. We hope by providing a user-friendly GUI application, tourists can easily find top-rated restaurants that match their location, desired cuisine, and other preferences.

2. Implementation Details

2.1 Technologies:

- **Backend Language:** Python
- **Data Source:** Yelp Open Dataset

- **BM25 Ranking:** Reviews are analyzed for positive keywords like "excellent" and "delicious".Businesses are ranked based on how well their reviews match these keywords using the BM25 algorithm.

2.2 Functionality:

- **User Input:**

  Keywords (we recommend using words with underline for testing):

  Users can provide any keyword related to their desired dining experience.This allows for flexible search based on various criteria, including:

  - Food category: Italian, Mexican, Thai, etc.
  - Price range: Cheap, moderate, expensive, etc.
  - Ambiance: Romantic, casual, family-friendly, etc.
  - Specific features: Waterfront views, live music, outdoor seating, etc.
  - Cuisine preferences: Vegan, gluten-free, vegetarian, etc.
  - Other desired attributes: Good for groups, late-night dining, etc.

  Location:
  Users can choose one of two options:

  - Current location: The application will automatically use the user's device location to find nearby restaurants.
  - Specific location: Users can manually enter a city, address, or landmark to search for restaurants in a specific area, for example, New York, Jersey City

- **Yelp API Integration:**

  - Queries the Yelp Open Dataset based on user input.
  - Filters restaurants based on BM25 score and combines it with other factors like user reviews and keyword relevance .
  - Retrieves relevant information, including restaurant name, cuisine, ratings, and location.

- **Main functions**
  - `get_address_by_ip()`
    Retrieves user's approximate location based on their IP address. User provided IP address and return latitude and longitude coordinates if successful.

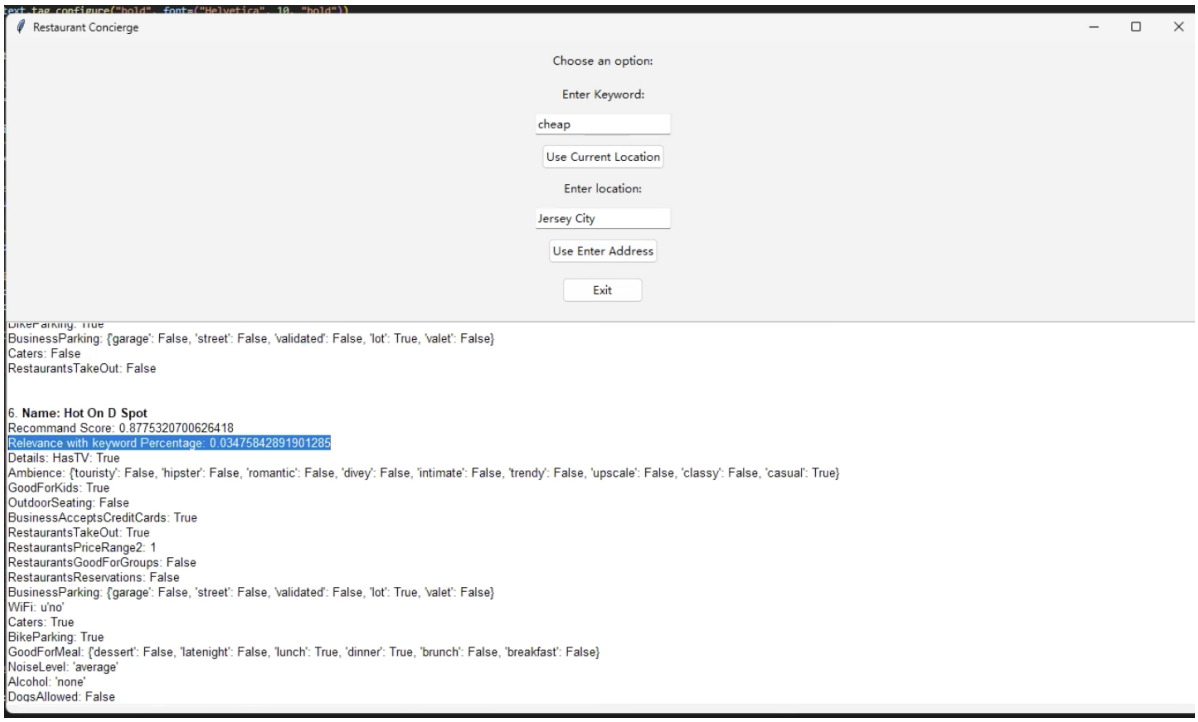
- `get_business_within_distance(data, user_location)`
  Get lat long from the data set for each business, calculate the distance for the address, passing distance is the distance we want to filter, and user_location = [lat,long] that user is located.

- `get_rank(keyword, weight_bm25=0.7, weight_useful=0.33, weight_funny=0.05, weight_cool=0.05)`
  The `get_rank` function utilizes an algorithm that calculates the ranking score based on processed queries. It applies the BM25 formula to obtain the score, combining it with review stars, useful count, funny count, and cool count, each assigned a different weight.The query holds the most weight, while the others have varying weights. Additionally, the function returns the `relevance_percentage` based on how frequently the keyword appears in the review.

- **Restaurant Recommendation:**

  - Generates a curated list of top-rated restaurants based on user input and filtered data.

2.3 User Interface: Simple graphical user interface **(GUI)**



2.4 Future Enhancements (Stretch Goals):

- Budget considerations
- Party size
- Reservation information
- Optimal Runtime (algorithm efficiency)

3. Code Repository

https://github.com/rheaxychen/CourseProject

4. Usage Documentation

4.1 Prerequisite

- Env requirement: python 3.10 (3.7 and higher might be fine too)
- Hardware requirement:
    - Intel based (M1/M2 will run into problems opening GUI)
    - Memory 16GB or more (less will result in extra wait time and potential problem loading data/GUI)

4.2 Installation

- Download Yelp's open dataset from [Yelp Dataset] and extract it.
- Put all the yelp data files inside the folder ./yelp_dataset/*, name should be yelp_academic_dataset_*.json such like that
- Install required packages : `pip install -r requirements.txt`

4.3 Running the Application

```
python restaurant_concierge_gui.py
```

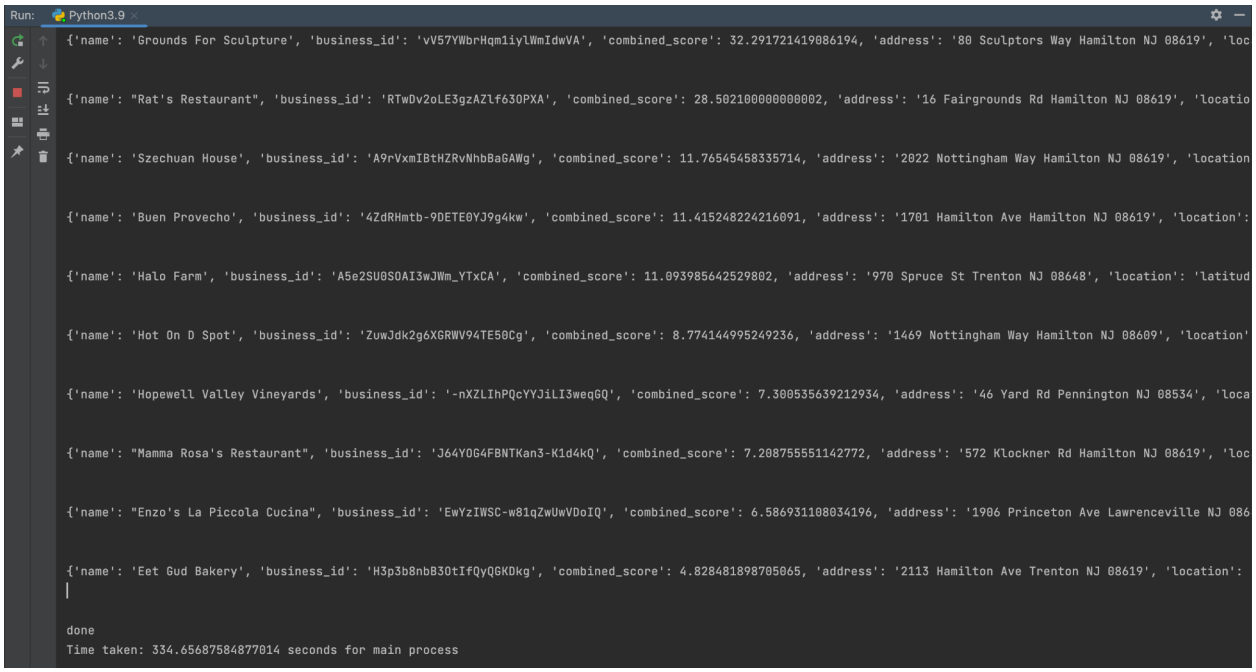
4.4 User Input:

Keywords, Location (See the second part “2.2 User Input” for details.)

```
Choose an option to get recommendations:
1. Using location from the current address
2. Inserting an address
0. Exit
Enter the number corresponding to your choice: 2
Enter the address you want to search: New York
```

4.5 Output:

A list of top restaurants based on the user's input and preferences. This list will include the restaurant's name, score, price range,parking information, etc.



5. Demo

This video will guide you through installation, setup and usage.

https://mediaspace.illinois.edu/media/t/1_x9km4tge

6. Evaluation

The application's success will be evaluated based on the following criteria:

- **Relevance of Recommendations:**
  - Restaurants should match user preferences and location constraints.
  - Recommendations should offer a variety of options based on user input.
- **User Satisfaction:**
  - User feedback will be collected to gauge satisfaction with the application's functionality and usability.
  - User surveys or tests may be conducted to gather insights for improvement.
- **Ease of Use:**
  - The GUI interface should be user-friendly and intuitive.
  - Navigation and interaction with the application should be simple and efficient.

7. Conclusion

This project aims to develop a user-friendly and efficient restaurant concierge service for tourists with limited time. By leveraging the Yelp Open dataset and a user-centered design, the application offers a streamlined solution for finding top restaurants and maximizing dining experiences during short trips. The project will be evaluated based on user feedback and its ability to generate relevant and satisfying recommendations.

8. Team member contribution

Rhea Chen (captain):
Help with design, submit deliverables, track progress, and present final project demo
Jingjing Yao:

Help with design, make final project documentation

Xian Chen:

Design and implementation