Webscraping and Recommendation System for Paris Restaurants

Mael DARNAUD
Stella HU
Hugo LABORDE

Contents

1 Introduction

This project focuses on collecting detailed information about restaurants in Paris to develop a personalized restaurant recommender system for tourists and locals. The data was collected using two webscraping approaches: leveraging the Tripadvisor API and scraping Google Maps with Selenium. The goal was to ensure diverse, high-quality data for creating an accurate and user-friendly application.

2 Webscraping Methods

2.1 Tripadvisor API

The Tripadvisor API was employed as a primary data source. Three specific functions were utilized to gather information about restaurants:

- **Nearby Search**: This function retrieves up to 10 restaurants located near a given latitude and longitude. To ensure diverse coverage, random GPS points within Paris were generated, enabling us to avoid Tripadvisor's bias toward showing only the most popular restaurants.
- **Location Details**: This function returns detailed information for a specific location, including the name, address, rating, and URL for the listing on Tripadvisor. It was used to gather key attributes of each restaurant.
- **Location Reviews**: This function retrieves up to 5 of the most recent customer reviews for a restaurant. This allowed us to include customer opinions in our dataset.

The use of these functions ensured that the dataset contained both comprehensive details about each restaurant and real customer feedback.

2.2 Google Maps Scraping with Selenium and BeautifulSoup

In addition to the Tripadvisor API, we used Selenium and BeautifulSoup to scrape data from Google Maps. The scraping process was conducted as follows:

- Searches for restaurants were performed for each of the 20 districts (*arrondissements*) of Paris.

- Details such as restaurant name, address, type of cuisine, average price, and reviews were collected and added to the dataset.

This method complemented the Tripadvisor data by adding restaurants that might not have been listed on Tripadvisor, ensuring a more diverse dataset.

2.3 Data Consolidation

The datasets collected from the Tripadvisor API and Google Maps scraping were concatenated into a single comprehensive dataset. This dataset served as the foundation for building the recommendation system.

3 Data Overview

The final dataset included the following attributes:

- **Name**: The name of the restaurant.
- **Description**: A brief description of the restaurant.
- **Cuisine Type**: The type(s) of cuisine served (e.g., Italian, French, etc.).
- **Rating**: The average user rating on a scale of 1 to 5.
- **Opening Hours**: The operating hours of the restaurant.
- **Location**: The address or GPS coordinates of the restaurant.
- **Average Price**: The estimated average cost of a meal in euros (€).
- **Reviews**: Customer reviews gathered from Tripadvisor and Google Maps.

4 Application Development

The collected data was used to develop a Streamlit-based application called the **Restaurant Recommender**. The application offers the following features:

- Personalized restaurant suggestions based on user input (e.g., preferred cuisine, budget, or location).
- Filters to sort results by price range and district (*arrondissement*) within Paris.
- Recommendations ranked using a weighted scoring system that considers price-to-quality ratios and semantic similarity.
- A simple, user-friendly interface powered by Streamlit.

The application uses precomputed embedding vectors for restaurant descriptions, enabling efficient and accurate similarity matching. The recommendation engine is driven by advanced machine learning techniques to match user preferences with restaurant options.

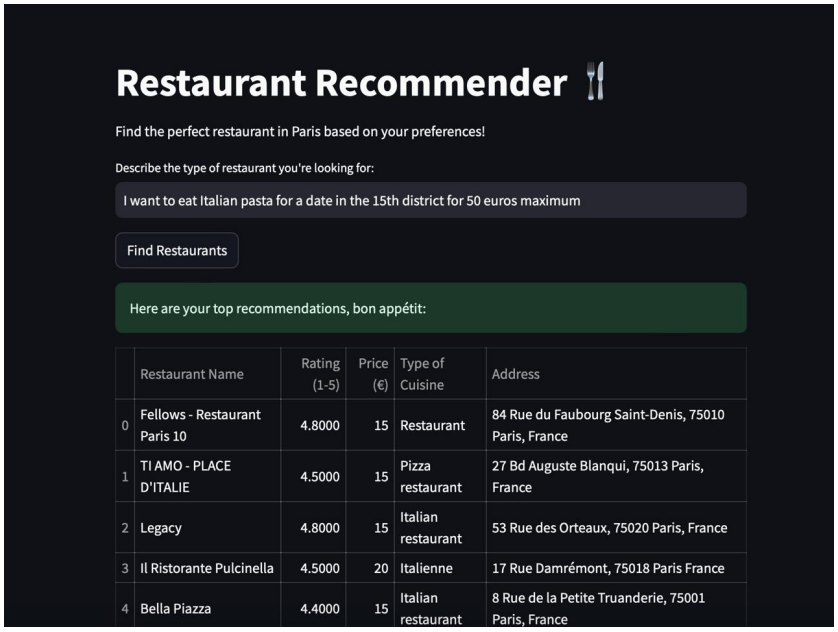

Figure 1: Example of a research on the App

5 How to Use the Application

To use the application, follow these steps:

1. Place the ‘restaurants.csv‘ dataset (containing the collected data) in the same directory as the application.

2. Run the Streamlit application using the following command:

```
streamlit run app.py
```

3. Enter your preferences in the input box, such as ”a cozy French bistro in the 5th arrondissement with a budget-friendly price.”

4. The application will display the top 5 recommended restaurants based on your input.

6 Conclusion

This project demonstrates how webscraping and machine learning can be combined to create a practical and user-friendly recommendation system. By leveraging multiple data sources and integrating diverse data points, the Restaurant Recommender provides valuable insights and recommendations for both tourists and locals in Paris. The application’s flexibility and robust design make it a useful tool for exploring the rich culinary landscape of Paris.