Welcome to the Zomota Cuisine Recommender project repository! This project aims to build a smart cuisine recommendation system using machine learning and collaborative filtering techniques. The system is designed to suggest various cuisines to users based on their preferences, previous orders, and similar users' preferences. This README file will guide you through the project and provide instructions on how to set up and use the recommender system.

Table of Contents

Project Overview

Getting Started

Installation

Dependencies

Data Collection and Preprocessing

Collaborative Filtering Algorithm

User Interface

Evaluation Metrics

Contributing

License

1. Project Overview

The Zomota Cuisine Recommender project is an intelligent system that takes into account users' historical order data and provides personalized cuisine recommendations. The main components of the project include data collection, preprocessing, collaborative filtering algorithm, and a user interface for interacting with the recommendation system.

2. Getting Started

To get started with the project, follow these steps:

Installation

Clone the project repository to your local machine:

bash

Copy code

git clone https://github.com/your-username/zomota-cuisine-recommender.git

Dependencies

Make sure you have the following dependencies installed:

Python 3.x

Pandas

NumPy

Scikit-learn

Flask (for the user interface)

Jupyter Notebook (optional, for exploring data and models)

You can install the Python dependencies using pip:

Copy code

pip install pandas numpy scikit-learn flask

3. Data Collection and Preprocessing

The data used for training and testing the cuisine recommender system should be in a structured format. Typically, this data includes information about users, their historical orders, and corresponding cuisines.

Ensure that you have the necessary data files in the correct format. You might need to preprocess the data to remove duplicates, handle missing values, and convert it into a suitable format for model training.

4. Collaborative Filtering Algorithm

The heart of the cuisine recommender system lies in the collaborative filtering algorithm. This algorithm analyzes user preferences and identifies similarities between users to make recommendations. The code for the collaborative filtering algorithm can be found in the collaborative\_filtering.py file. You can customize the algorithm and its parameters as needed.

5. User Interface

To provide a user-friendly experience, this project includes a web-based user interface using Flask. The user interface allows users to input their preferences, and the system responds with cuisine recommendations. The Flask application can be found in the app.py file.

To run the user interface, execute the following command:

Copy code

python app.py

The application will be accessible at http://localhost:5000 in your web browser.

6. Evaluation Metrics

To assess the performance of the cuisine recommender system, you can use various evaluation metrics such as precision, recall, F1 score, or Mean Absolute Error (MAE). These metrics help to measure how accurate and reliable the recommendations are.

7. Contributing

We welcome contributions to improve the project. If you have any bug fixes, feature enhancements, or new ideas, feel free to submit a pull request. Please follow the standard coding conventions and include clear commit messages.