Food Recommendation System Project Report

1. Project Overview

The Food Recommendation System uses machine learning techniques to predict the ratings of various food items and recommend personalized items to users based on their preferences. The system uses a dataset of food responses, and the goal is to predict food ratings based on categorical and numerical features like Food Court/Cafe Location, Cuisine Type, Veg/Non-veg, Meal Type, Spice Level, and Price.

2. Data Overview

- Dataset: food_responses.csv
- Features:
  - Categorical Features: Food Court/Cafe Location, Cuisine Type, Veg/Non-veg, Meal Type, Spice Level
  - Numerical Feature: Price
  - Target: Rating
  - Additional: Timestamp (removed during data processing)

3. Preprocessing Steps

- Data Loading: The dataset is loaded using pandas.read_csv.
- Data Cleaning: The Timestamp column was dropped, as it is not relevant to the recommendation model.
- Feature Engineering:
  - Categorical features are one-hot encoded using OneHotEncoder from sklearn to convert them into numerical representations.
  - Categorical features include Food Court/Cafe Location, Cuisine Type, Veg/Non-veg, Meal Type, and Spice Level.
- Target Variable: Rating, which is the target for prediction.

4. Model Implementation

- Model: A RandomForestRegressor is used to predict food ratings based on the features.
- Libraries Used:
  - Pandas: For data manipulation and processing.
  - Scikit-learn: For machine learning algorithms and preprocessing.
  - NumPy: For numerical operations.

Steps in Model Implementation:

1. Data Preparation:
   - Categorical features are encoded using OneHotEncoder.
   - The encoded categorical features are then concatenated with numerical features (Price).

2. Model Training:
   - A RandomForestRegressor model is trained with 100 estimators, using the features (X) and target (y).
   - The model learns the relationship between the input features and the target variable (Rating).

3. Prediction:
   - After training, the model predicts ratings (Predicted_Rating) for each food item in the dataset.

5. Evaluation Metrics

The following evaluation metrics are used to assess the performance of the model:

1. R^2 (R-squared): Measures the proportion of the variance in the target variable that is explained by the independent variables.
   - Ideal Value: Closer to 1 is better. For example, values above 0.7 are generally considered good.
   - Our Value: 0.8114

2. RMSE (Root Mean Squared Error): Measures the average magnitude of the errors in predictions. It penalizes larger errors more due to squaring.
   - Ideal Value: Lower values are better. A value below 10% of the range of the target variable is good.
   - Our Value: 0.5545

3. MAE (Mean Absolute Error): Measures the average absolute differences between the predicted and actual values.
   - Ideal Value: Lower values are better, and a value less than 10% of the target variable's range is desired.
   - Our Value: 0.3903

6. Recommendation Logic

The system recommends food items based on user preferences for various factors (e.g., Food Court/Cafe Location, Veg/Non-veg, Price, Spice Level). The process follows these steps:

1. Input: User inputs preferences like Food Court Location, Cuisine Type, Spice Level, Price Range, and Rating.
2. One-hot Encoding: The user's categorical inputs are encoded using the same encoder that was trained on the dataset.
3. Cosine Similarity: A similarity score is computed between the user input and all food items in the dataset using cosine similarity.
4. Filtering: The system filters food items based on the user's input criteria (e.g., matching Spice Level, Price, Rating, etc.).
5. Sorting: The recommendations are sorted based on the cosine similarity score, predicted rating, and spice match.
6. Output: The top 5 recommended food items are returned.

7. Streamlit Interface

The Streamlit interface allows users to input their preferences and view the recommended food items. The interface includes sliders for Price and Rating selection, as well as dropdowns for Food Court Location, Cuisine Type, Meal Type, Veg/Non-Veg, and Spice Level. Upon selecting these options, the system displays the top 5 food recommendations.

8.Conclusion

The Food Recommendation System successfully predicts ratings and recommends food items based on user preferences. The system uses RandomForestRegressor for prediction and cosine similarity for generating personalized recommendations. The evaluation metrics indicate that the model performs well in terms of explaining the variance (R^2), and the errors (RMSE and MAE) are acceptable, indicating good prediction accuracy.

Future Improvements

- Model Optimization: The model could be fine-tuned using hyperparameter optimization techniques such as grid search or random search.
- Additional Features: More features, such as customer reviews or historical data, could be incorporated to improve recommendation accuracy.
- Real-time Data: Incorporate real-time updates for food availability or dynamic pricing in the recommendation system.

GITHUB Link

https://github.com/akshatangi/Food_Recommender/

A Project By:

Akshat Kumar Angi, SAP: 500120178
Anirudh Rana, SAP: 500119643