Recipe Recommendation Model: Report

1. Problem Understanding

Objective

The objective of this project is to design and build a model that recommends food recipes based on input ingredients, dietary preferences, and cuisine styles. The goal is to leverage machine learning to create an efficient recommendation system that assists users in discovering recipes matching their preferences.

AI and Machine Learning in Recipe Creation

AI and machine learning offer significant innovations in recipe recommendation and creation:

- AI can study patterns in existing recipes to suggest ingredients that go well together.
- Machine learning models can customize recipe suggestions based on user preferences, dietary needs, and favorite cuisines.
- Generative AI models can create entirely new recipes by predicting the next step or ingredient based on a learned dataset.

Factors Influencing Recipe Recommendations

1. **Dietary Restrictions:** Avoiding allergens or non-preferred ingredients (e.g., gluten-free, vegan).
2. **Cuisine Preferences:** Filtering recipes by cuisine type.
3. **Recipe Complexity:** Matching suggestions to user preferences for prep/cook time or skill level.
4. **Nutritional Requirements:** Providing recommendations aligned with health goals like low-carb or high-protein diets.

2. Dataset Collection and Preparation

Dataset

The dataset used contains recipes sourced from publicly available datasets (e.g., Kaggle). Each recipe includes attributes such as name,

ingredients, directions, cuisine type, and additional metadata like prep time, cook time, and nutritional information.

Preprocessing Steps

Load and Inspect Dataset:

- Identified null values, duplicates, and irrelevant columns.
- Sample inspected to understand the structure.

Cleaned Data:

- Removed unnecessary columns (url, rating, nutrition, etc.).
- Normalized recipe names to lowercase and stripped whitespace.
- Extracted cuisine labels from the path column for categorical use.

Ingredient Normalization:

- Cleaned ingredient text using regular expressions to remove special characters, parentheses, and extra spaces.
- Created a cleaned_ingredients column for uniform processing.

Feature Extraction:

- Vectorized cleaned_ingredients using TF-IDF Vectorizer, converting textual data into numerical vectors suitable for similarity computation.

3. Model Development

Approach

- **Content-Based Recommendation Model:** Utilized cosine similarity on TF-IDF ingredient vectors to suggest recipes similar to a given input.
- **Model Workflow:**
  1. Input recipe name is matched against the dataset.
  2. Ingredient vectors are compared using cosine similarity to compute closeness.
  3. The top N recommendations are selected based on similarity scores.
  4. Optional filters like cuisine type are applied.

Implementation

- Used TF-IDF Vectorizer for feature extraction.
- Computed Cosine Similarity to measure similarity between recipes.
- Built a Flask web application to provide a user interface for querying recommendations.

4. Evaluation and Insights

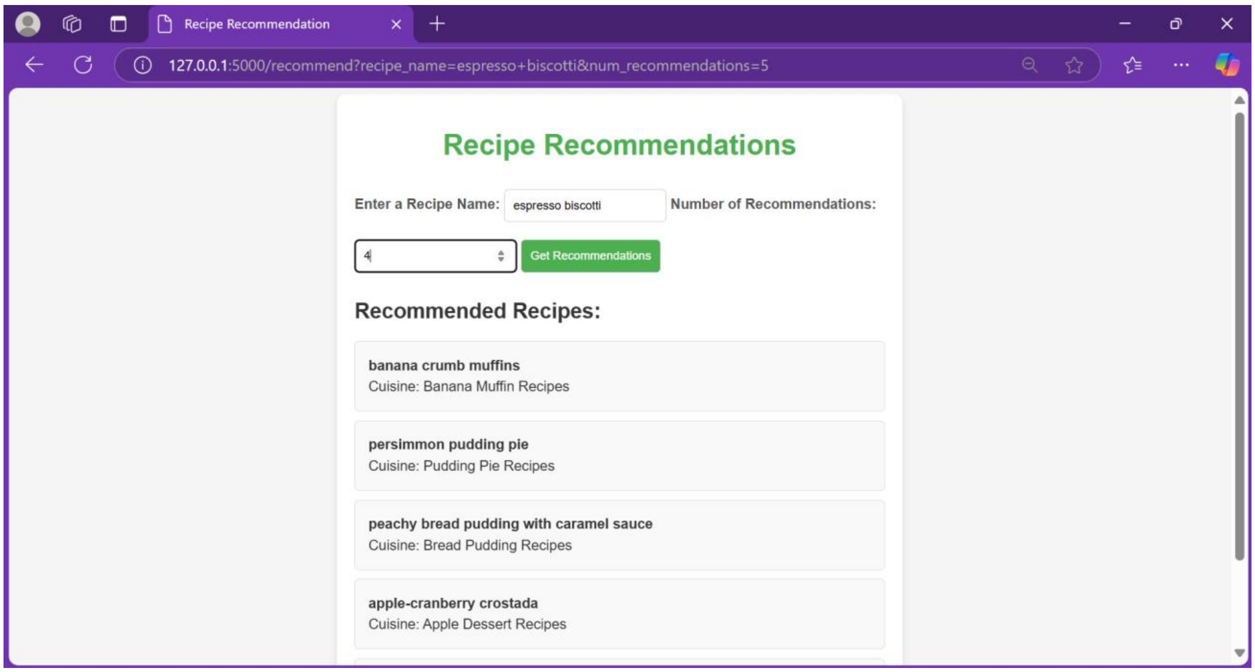
Evaluation Metrics

1. **Ingredient Coherence:** The recommended recipes were inspected to ensure that the suggested ingredients logically fit together.
    - Example Query: "espresso biscotti"
    - Recommendations included "banana crumb muffins", "persimmon pudding pie", and "apple-cranberry crostada", all logically related based on ingredients.
2. **Recipe Diversity:** Recommendations spanned multiple recipe types and cuisines, ensuring a varied output.
3. **User Testing:** Tested by querying multiple recipes and validating the quality of recommendations.

Performance

- The content-based filtering model performed well, with coherent and diverse suggestions for most queries.

5. Bonus: User Interface



A simple Flask-based web application was built to provide a user-friendly interface:

1. Users can input a recipe name and specify the number of recommendations.
2. Optional filters for cuisine type.
3. Results are displayed in a clean, styled HTML interface.

6. Limitations and Future Improvements

Limitations

1. The model depends heavily on the quality and uniformity of the dataset.
2. Recommendations are limited to the recipes present in the dataset.
3. No generative capabilities (e.g., creating entirely new recipes).

Future Improvements

1. **Incorporate Generative Models:** Use GPT-based architectures to generate new recipes based on input ingredients.
2. **Personalized Suggestions:** Customize recommendations based on user preferences.
3. **Nutritional Information:** Provide health details like calories, nutrients, and portion sizes.
4. **Ingredients Search:** Let users find recipes by directly entering available ingredients.