Meal Harmony: Using Machine Learning to simplify food choices

Prepared by:  
Paarth Jha (2021BTech080)  
Mayank Bohra (2021BTech072)  
Manasvi Jain (2021BTech070)

Faculty Guide:  
Dr. S Taruna



Institute of Engineering and Technology  
JK Lakshmipat University, Jaipur  
  
May 2024

Index

|  |  |  |
| --- | --- | --- |
| 1 | Abstract | 3 |
| 2 | Introduction | 4 |
| 3 | Literature Review | 5 |
| 4 | Problem Statement | 6 |
| 5 | Research | 7 |
| 6 | How it works | 10 |
| 7 | Demonstration | 11 |
| 8 | Future Plan | 12 |
| 9 | Conclusion | 13 |
| 10 | References | 14 |

Abstract

Introduction

In today's fast-paced world, deciding what to cook for meals can often feel like a daunting task for many households. The familiar refrain of "What should we make?" followed by a vague "Anything" is all too common. This indecision often leads to repetitive meals or unhealthy instant options, ultimately impacting the well-being of family members. Moreover, factors like individual preferences, health conditions, and ingredient availability further complicate the decision-making process, leaving many families feeling overwhelmed.

Recognizing the need for a solution to this everyday challenge, our project sets out to develop a personalized food recommendation system. By harnessing the power of machine learning, particularly deep learning and statistical methods, we aim to change the way families plan their meals. This system will take into account not only the unique tastes and dietary requirements of each family member but also external factors such as weather conditions and ingredient availability to offer tailored meal suggestions.

At its core, our project seeks to streamline the meal planning process, making it easier, faster, and healthier for households. By using machine learning algorithms and data analysis techniques, we aim to enable families to make informed decisions about what they eat, ultimately promoting better health and well-being for all. Through this innovative approach, we envision a future where the age-old question of "What should we cook?" is met with confidence and excitement rather than uncertainty and stress.

The motivation for this project comes from our own homes, where our mothers constantly ask, “What do you want to eat today?”, and we reply with, “Anything”.

Literature Review

Saito, Asada, Yoshitomi, Kato, and Tabuse [1] in their paper, presented the development of a recipe recommendation system using collaborative filtering and impression words. The system utilizes MMDAgent (MikuMikuDanceAgent) as the user interface for the application. Their suggested recommendation process involves two steps: the first step terminates based on a previously decided condition, and the second step identifies the recipe most like past successful recommendations. Their system can narrow down potential recipes to a single choice. The system proposed by them was evaluated through a questionnaire survey of 10 user participants regarding the recommendations produced by the system. The evaluation includes four categories: recommendation accuracy, user satisfaction, usefulness of impression words, and usefulness of collaborative filtering. The study proposes considering yesterday's dinner recipe as a factor in narrowing down recipe choices to a final recommendation, indicating the potential for further exploration in this area.

Khang and Cheung [2] in their paper discussed the use of different deep learning approaches for sentiment analysis, automatic review tag generation, and retrieval of food reviews. Their proposed sentiment analysis techniques use LSTM, RNNs and BERT deep learning models for analyzing the sentiment of food reviews. The paper also presents a Part-of-Speech (POS) algorithm for automatically identifying and extracting adjective-noun pairs from the review content for review tag generation. In terms of retrieval of food reviews, the paper discusses the use of a Solr-based search system and the integration of a RankNet model for re-ranking the retrieval results. The experimental results of their work showed the promising performance of the proposed deep learning approaches in addressing real-world problems related to food reviews and recommendations.

Also, Singh and Dwivedi [3] in their paper, proposed a food recommendation system using content-based and collaborative filtering techniques. They utilized K-nearest neighbour methods and a food dataset obtained from Kaggle. The system recommends food, based on attributes such as food name, food id, cuisine type, and diet type (veg or non-veg) for content-based filtering. Collaborative filtering is used with attributes such as user id, food id, and rating to recommend food. The performance of the proposed model is evaluated using error values like root mean square error (RMSE) and mean absolute error (MAE) to quantify the accuracy of the recommendations. The recommendations are sorted based on average ratings and food names, ensuring that the most highly rated and relevant food items appear at the top. The system incorporates both numerical ratings and textual information to make personalized recommendations.

Adomavicius & Tuzhilin [4] in their paper provided an overview of their current generation of recommendation methods, including content-based, collaborative, and hybrid approaches. It discusses the limitations of current recommendation methods and proposes extensions to improve recommendation capabilities, such as incorporating contextual information and supporting multi-criteria ratings. Beyond traditional movie recommendations, more advanced recommendation methods are emphasized.

Problem Statement

This project deals with a common problem in households: figuring out what to cook for meals. Often, people end up making the same old dishes or something quick but not very healthy. Sometimes, they spend a lot of time deciding what to cook. And even when they decide, there are factors like health conditions, weather, and ingredient availability to consider. This project aims to solve this problem by creating a system that suggests personalized meal options using advanced computer technology. By understanding each person's likes, this system suggests the best meals for everyone in the family.

Research and Methodology

1. Content-Based Filtering

Content-based filtering is a recommendation system technique that suggests items to users based on the features or characteristics of the items themselves. It focuses on analyzing the properties of items rather than user-item interactions.

In content-based filtering, items are represented by a set of features or attributes. These features could include textual descriptions, metadata, or other relevant information about the items. For example, in a movie recommendation system, features of movies could include genre, director, actors, and plot summary.

The recommendation process begins by creating user profiles based on their preferences. These profiles are built using the features of items the user has interacted with or rated positively in the past. Then, the system identifies items that are similar to those liked by the user based on the features. This is typically done using similarity measures such as cosine similarity or Euclidean distance.

Content-based filtering is particularly useful in scenarios where explicit user feedback (like ratings) is sparse or unavailable, as it relies solely on item attributes. However, it may suffer from the "filter bubble" problem, where users are recommended items similar to what they already like, potentially limiting serendipitous discovery. Additionally, it requires rich and accurate item feature representation to be effective.

1. User-Item Interaction Matrix

A user-item interaction matrix is a fundamental data structure used in recommendation systems to represent interactions between users and items. It is essentially a two-dimensional matrix where one dimension represents users, and the other represents items.

Each cell in the matrix corresponds to a user-item interaction, typically denoting a user's behavior towards an item. This behavior could be explicit, such as ratings or reviews given by users to items, or implicit, such as clicks, views, purchases, or time spent on items.

The matrix is usually sparse because not every user interacts with every item, and users often interact with only a small subset of available items. For example, in a movie recommendation system, the matrix might have users as rows and movies as columns. The cells of the matrix would contain ratings given by users to movies they have watched, with many cells remaining empty as users haven't rated all movies.

The user-item interaction matrix serves as the basis for various recommendation algorithms, including collaborative filtering and content-based filtering. It allows recommendation systems to understand user preferences and make personalized recommendations by analyzing patterns of interactions between users and items. Additionally, it provides a convenient representation for modeling and analyzing user behavior in recommendation tasks.

1. Similarity Matrix

A similarity matrix represents the pairwise similarity between items or users in a recommendation system. Each cell (i, j) in the matrix holds a similarity score indicating how similar item i is to item j (or how similar user i is to user j). Similarity measures commonly used include cosine similarity, Pearson correlation coefficient, or Jaccard similarity. These measures quantify the resemblance between items or users based on their interactions or features. The similarity matrix serves as a fundamental data structure for collaborative filtering methods. It enables the identification of items (or users) that are most similar to each other, facilitating the recommendation process.

1. Neighbourhood Matrix

A neighborhood matrix, often derived from a similarity matrix, identifies a subset of similar items or users, known as the neighborhood, for each item or user in the system. For a given item or user, the neighborhood matrix contains a list of other items or users that are deemed most similar, based on the similarity scores in the similarity matrix. The size of the neighborhood typically depends on a predefined parameter, such as the number of nearest neighbors or a similarity threshold. Neighborhood matrices play a crucial role in collaborative filtering algorithms by enabling personalized recommendations. By considering the preferences of similar users (or items) in the neighborhood, the system can infer the preferences of the target user and recommend relevant items accordingly.

1. Linear Regression

Linear regression is a statistical method used to model the relationship between one or more independent variables and a dependent variable. It assumes a linear relationship between the independent variables (often denoted as 𝑋) and the dependent variable (often denoted as 𝑌).

In simple linear regression, there is only one independent variable, while in multiple linear regression, there are multiple independent variables.

The goal of linear regression is to estimate the coefficients that represent the relationship between the independent variables and the dependent variable. These coefficients are adjusted to minimize the difference between the observed values of the dependent variable and the values predicted by the model.

Linear regression can be used for prediction, where given the values of the independent variables, it predicts the value of the dependent variable. It can also be used for inference to understand the relationship between the independent and dependent variables and identify significant predictors of the outcome.

Linear regression is widely used in various fields, including economics, finance, social sciences, and machine learning, due to its simplicity and effectiveness in modeling linear relationships between variables.

1. Pandas Library

Pandas is a powerful Python library for data manipulation and analysis. It provides data structures and functions to efficiently work with structured data, such as tables and time series. Pandas' main data structure is the DataFrame, which is a two-dimensional labeled data structure with columns of potentially different data types. It offers functionality for data cleaning, filtering, merging, and reshaping, as well as statistical operations and time series analysis. Pandas simplifies data handling tasks, making it a go-to tool for data scientists, analysts, and developers working with tabular data in Python.

1. NumPy Library

NumPy is a fundamental Python library for numerical computing. It provides powerful tools for creating and manipulating arrays and matrices, which are essential data structures in scientific computing and data analysis. NumPy's ndarray (N-dimensional array) enables efficient operations on large datasets, including mathematical, logical, shape manipulation, sorting, and statistical functions. With its broad range of mathematical functions, NumPy facilitates complex mathematical operations and linear algebra computations. Its high performance and memory efficiency make it a cornerstone of many Python scientific computing ecosystems. NumPy is widely used in fields such as machine learning, signal processing, physics, and engineering.

1. Scikit-Learn Library

Scikit-learn is a versatile Python library for machine learning. It provides simple and efficient tools for data mining and data analysis, built on NumPy, SciPy, and matplotlib. Scikit-learn offers a wide range of algorithms for classification, regression, clustering, dimensionality reduction, model selection, and preprocessing. It features a consistent API, making it easy to experiment with different algorithms and workflows. With extensive documentation and community support, scikit-learn is suitable for both beginners and experienced practitioners in machine learning. It's widely used in academia and industry for developing and deploying machine learning models across various domains, from healthcare to finance and beyond.

**Challenge**  
The biggest challenge we had was data collection. A food recommendation system like this was being worked upon by no one, hence required data was not available. We took this opportunity to create a Google Form to help us collect the data. We received 46 responses, but while cleaning and wrangling we had to remove many due to incomplete choices. In the end, we made the model using 36 responses.

How it works

Our recommendation system operates on the principle of collaborative filtering, a popular technique in recommendation systems that leverages the preferences and behaviors of similar users to make recommendations. The system takes a user ID as input and prompts the user to select a dish from a predefined list. Upon selection, the system updates the user-item interaction matrix to reflect the user's preference for the chosen dish. Additionally, the system offers recommendations based on the user's past interactions and the similarity between dishes.

**Data Representation:**

The core of our recommendation system revolves around the user-item interaction matrix, a tabular representation of users' preferences for various dishes. Each row in the matrix corresponds to a user, while each column represents a dish. The values in the matrix indicate the degree of interaction between users and dishes, with higher values indicating stronger preferences.

**Model Training:**

To initialize the recommendation system, we employ a machine learning model, specifically linear regression, to predict user preferences based on historical data. The model learns the relationship between user attributes and dish preferences, enabling it to make informed predictions about users' preferences for new dishes.

**User Interaction:**

Upon receiving a user ID, the recommendation system prompts the user to select a dish from a predefined list of options. The user's selection serves as feedback to update the user-item interaction matrix, reflecting the user's preference for the chosen dish. Additionally, the system offers the option to receive recommendations based on the user's past interactions.

**Recommendation Generation:**

When the user selects the option to receive recommendations (indicated by inputting 111), the system generates recommendations based on the user's past interactions and the similarity between dishes. Specifically, the system computes the similarity matrix, which quantifies the similarity between pairs of dishes based on user interactions. Using this similarity matrix and the user's interaction history, the system identifies the top dishes liked by the user and recommends them to the user.

**Updating the Model:**

To ensure the recommendations remain relevant and up-to-date, the recommendation system dynamically updates the user-item interaction matrix and the similarity matrix based on user interactions. When the user selects a dish, the system adjusts the interaction scores for that dish and its neighbors in the similarity matrix, reflecting the user's feedback. This iterative process enables the system to adapt to changes in user preferences and provide increasingly accurate recommendations over time.

Model Demonstration

A computer screen shot of a black screen

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screen shot of a computer

Description automatically generated

Future Plan

As we look ahead to the future, our vision for the food recommendation model encompasses a multifaceted approach aimed at further enhancing the user experience and delivering more personalized recommendations. By leveraging advanced techniques such as sentiment analysis and incorporating temporal dynamics, we aim to create a more refined and context-aware recommendation system that caters to the evolving preferences and behaviors of our users.

One of the key pillars of our future plan involves the collection of organized and relevant data pertaining to user preferences, dish attributes, and contextual factors. By gathering data from diverse sources such as user reviews, nutritional information, and seasonal trends, we aim to enrich our dataset and capture a comprehensive understanding of the food landscape.

To augment the recommendation model, we propose the integration of sentiment analysis techniques to extract deeper insights from user-generated content. By analyzing the sentiment expressed in user reviews, comments, and social media interactions related to specific dishes, we can discern the emotional responses and subjective preferences associated with each dish. This sentiment analysis will inform the recommendation process, enabling us to recommend dishes that resonate positively with users on an emotional level.

Recognizing the influence of temporal dynamics and seasonality on food preferences, our future plan involves incorporating time-sensitive factors into the recommendation model. By considering temporal trends, such as seasonal ingredients, cultural festivities, and culinary trends, we can offer timely recommendations that align with users' current preferences and dietary inclinations. Additionally, we will explore the incorporation of user-specific temporal patterns, such as mealtime preferences and weekday versus weekend dining habits, to further tailor recommendations to individual users.

The implementation of these enhancements will be undertaken as part of a major project in the upcoming semester. Our interdisciplinary team will collaborate closely to design, develop, and deploy the enhanced food recommendation model, leveraging expertise in data science, natural language processing, and user experience design. Through iterative development cycles and rigorous testing, we will refine the model to ensure its effectiveness, scalability, and usability.

After thorough testing and modifications, the team will create a mobile application and deploy the model locally for alpha testing, and online for beta testing.

Conclusion

In this report, we have explored the design and implementation of a dynamic food recommendation system that leverages collaborative filtering techniques to provide personalized dish recommendations based on user interactions. By harnessing user-item interaction matrices, machine learning models, and similarity metrics, our recommendation system delivers tailored recommendations that enhance user satisfaction and engagement.

Key components of our recommendation system include the user-item interaction matrix, which captures users' preferences for various dishes, and the linear regression model, which predicts user preferences based on historical data. Through iterative updates to the interaction matrix and the incorporation of similarity metrics, our system adapts to changes in user preferences and delivers increasingly accurate recommendations over time.

Looking ahead, our future plan involves the integration of advanced technologies such as sentiment analysis and the consideration of temporal dynamics to further refine the recommendation process. By analyzing sentiment in user-generated content and incorporating temporal factors such as seasonality and mealtime preferences, we aim to create a more sophisticated and context-aware recommendation system that caters to the evolving needs and preferences of our users.

The implementation of these enhancements will be undertaken as part of a major project in the upcoming semester, with our interdisciplinary team collaborating to design, develop, and deploy the enhanced recommendation model. Through iterative development cycles and rigorous testing, we will refine the model to ensure its effectiveness, scalability, and usability.

In conclusion, our recommendation system represents a significant step forward in the realm of food discovery, offering users a personalized and enriching experience that transcends traditional recommendation approaches. With our commitment to innovation and user-centric design, we are poised to revolutionize the way users explore and enjoy culinary delights, providing a seamless and delightful food discovery journey for users around the globe.

References

* Saito, K., Asada, T., Yoshitomi, Y., Kato, R., & Tabuse, M. (2018). A Recipe Decision Support System Using Knowledge Information and Agent. J. Robotics Netw. Artif. Life, 5(3), 204-207.
* Khang Le, T., & Cheung Hui, S. (2022). Machine Learning for Food Review and Recommendation. arXiv e-prints, arXiv-2201.
* Singh, R., & Dwivedi, P. (2023, July). Food Recommendation Systems Based on Content-based and Collaborative Filtering Techniques. In 2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT) (pp. 1-5). IEEE.
* Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and extensions. IEEE transactions on knowledge and data engineering, 17(6), 734-749.