Project Design

Yuteng Yang z5458030

1. Scope

1.1 Domain and Target Users

1.1.1 Domain

Home cooking recipe recommendation

1.1.2 Target Users

Amateur cooking enthusiasts, including both kitchen beginners (who heavily rely on recipes) and experienced home cooks (who seek innovation and healthy combinations). Users primarily access RecipeMate via mobile apps (Android/iOS).

1.2 Number of Recommendations & User Interface

1.2.1 Number of Recommended Items

6–8 recipes per session, ensuring diversity without overwhelming the user.

1.2.2 UI Design (Mockup)

- **Home Feed**: Users scroll vertically or swipe horizontally to browse recipe cards, each displaying an image, title, rating, cooking time.
- **Bottom Navigation Bar**: Includes three main modules – "Today's Recommendations", "My Favorites", and "Search".
- **Detail Page**: Tapping a card opens a detailed view with steps, required ingredients, nutritional info, and user reviews.
- **Feedback Controls**: "Like / Not Interested" and "Cooked" buttons are available on both cards and detail pages.

1.3 User Interaction & Feedback

1.3.1 Explicit Feedback

- “Like” (thumbs up) and “Not Interested” (thumbs down) are used to train user preferences.
- “Cooked” records actual cooking actions, helping evaluate recipe practicality.

1.3.2 Implicit Feedback

In-app dwell time and sentiment of user comments are used as supplementary signals.

1.3.3 Feedback Integration

All interactions are instantly uploaded to the backend and merged into the offline batch dataset; popular recipes get a temporary weight boost to balance popularity and personalization.

1.4 Dynamic Updates & Cold Start Strategies

1.4.1 Dynamic Model Updates

- Models are retrained offline daily at midnight, incorporating the latest user behaviors and newly added recipes.
- Online serving uses user embedding vectors and new recipe vectors for fast retrieval, with response times under 200ms.

1.4.2 New User Cold Start

- New users are prompted to specify cuisine preferences, taste profiles, dietary restrictions, and commonly used ingredients during registration.
- Initially, recommendations are purely content-based (TF-IDF text similarity + ingredient vectors), transitioning to hybrid recommendations once enough interaction data is collected.

1.4.3 New Recipe Cold Start

- Ingredient lists, cuisine tags, and textual descriptions are automatically extracted to generate content vectors for new uploads.
- Similarity with popular recipes is calculated to grant initial exposure.

1.5 Business Model & Monetization

- **Affiliate Marketing**: Recipe detail pages feature “One-Click Ingredient Purchase” links, integrating APIs from local supermarkets or e-commerce platforms, earning commissions on clicks or transactions.
- **Subscription Model**: Premium users receive personalized recipes from nutritionists, monthly health reports, and exclusive ingredient discount coupons.

- **Brand Collaborations**: Partner with cookware and food brands to showcase co-branded recipes or offer vouchers on the recommendation page.

2. Dataset(s)

2.1 Dataset Overview

https://www.kaggle.com/datasets/kanaryayi/recipe-ingredients-and-reviews

2.1.1 recipes.csv

- Size: Approximately 370,000 recipe records
- Key Fields: recipe_id, name/title, minutes, n_steps, ingredients, steps/directions, tags, description

2.1.2 reviews.csv

- Size: Approximately 2,000,000 user reviews
- Key Fields: review_id, user_id, recipe_id, rating, date, review_text

2.2 Field Usage

2.2.1 Content Features

- Ingredient vectors
- Text vectors (TF-IDF or pre-trained models)
- Tag information (multi-hot encoding)

2.2.2 Collaborative Filtering Features

- Rating matrix
- Implicit feedback (ratings or "Cooked" actions)

2.2.3 Temporal & Behavioral Features

- Timestamps
- Sentiment from review text

2.3 Data Preprocessing & Splitting

2.3.1 Missing Values & Outliers

- Remove recipes missing ingredients or steps
- Remove invalid or duplicate reviews

2.3.2 Data Splitting

- 8:1:1 split for train/validation/test
- Reserve no-history users for cold-start evaluation

2.4 Dataset Limitations & Challenges

- Sampling bias (Western cuisines)
- Cleaned data lacks real-world noise
- Static split doesn't match real-time update needs
- Missing ingredient availability/price info

2.5 Data Enhancement

- Integrate external APIs (Spoonacular, Edamam)
- User preference profiles
- Image-based features via CNN (if available)

3. Method(s)

3.1 Baseline: Content-Based Recommendation

3.1.1 Basic

Encode name + description + steps using TF-IDF or Sentence-BERT and represent ingredients as one-hot or embedding vectors.
Use cosine similarity with the user preference vector to recommend Top-N items.

3.1.2 Variants

1. Deep semantic encoding via pre-trained models
2. Feature fusion (text + ingredients)

3.1.3 Rationale

Works without rating history; ideal for cold-start; utilizes rich textual content.

3.2 Collaborative Filtering (CF)

3.2.1 Model-Based

Use SVD++ or NMF via the Surprise library on (user_id, recipe_id, rating)

3.2.2 Neighborhood-Based

- Item-Item CF
- User-User CF (k-nearest neighbors)

3.2.3 Rationale

Captures collective preferences; performs well for active users.

3.3 Hybrid Recommender System

3.3.1 Linear Weighted Combination

$$\boldsymbol{score(u,i) = \alpha \cdot CB_{score}(u,i) + (1 - \alpha) \cdot CF_{score}(u,i)}$$

3.3.2 Stacking Approach

Combine CB score, CF score, popularity, and sentiment into a meta-model.

3.3.3 Rationale

Balances content-based cold-start robustness with CF personalization. Flexible feature integration.

4. Evaluation

4.1 Offline Model Evaluation (Recommendation Model Metrics)

4.1.1 Accuracy Metrics

- Precision@6 / Recall@6
- NDCG@6

4.1.2 Diversity & Coverage

- Coverage
- Intra-List Diversity

4.1.3 Novelty & Serendipity

- Novelty
- Serendipity

4.1.4 Aggregate Metrics

- MAP
- ROC-AUC (for binary rating prediction)

We will prioritize **Precision@6** and **NDCG@6** as the primary metrics for offline model selection, as they balance recommendation accuracy and ranking relevance. **Diversity** and **novelty** will serve as secondary evaluation criteria to enhance user experience and prevent recommendation fatigue.

4.2 Online System Metrics

4.2.1 Interaction Metrics

- Click-Through Rate (CTR)
- Action Rate (e.g., "Cooked" or purchase)

4.2.2 User Behavior Metrics

- Session Depth
- Retention Rate (7-day, 14-day)

4.2.3 Business Metrics

- GMV
- Subscription conversion

4.3 User Study Design

4.3.1 Objective

Evaluate satisfaction, usability, and overall effectiveness.

4.3.2 Participants

20–30 home cooks (half beginners, half experienced)

4.3.3 Procedure

- Prototype demo via Figma/Flask
- Behavior logging (clicks, skips, etc.)
- 5-point Likert questionnaire:
    1. Recommendation satisfaction
    2. Perceived diversity
    3. Continued usage intention
- Semi-structured interviews

4.3.4 Analysis

- Quantitative: Survey results + behavioral logs
- Qualitative: Interview themes to inform iteration

4.4 Computational & Deployment Requirements

4.4.1 Real-Time Performance

- ⩽ 200ms response time
- Daily batch retraining

4.4.2 Resource Efficiency & Scalability

- Model microservices
- Vector caching

4.4.3 Monitoring & Rollback

- Real-time alerts on key metrics
- Version control and quick rollback