Hybrid Recommendation System for Mobiles

Sai Likhitha Bollu

M.Tech (Artificial Intelligence and Machine Learning)

VIT Vellore

Abstract

The contemporary mobile phone market is characterized by an overwhelming number of choices, leading to significant decision-making confusion for consumers. Traditional recommender systems, primarily based on Content-Based (CB) or Collaborative Filtering (CF), often fall short due to limitations like the cold-start problem, over-specialization, and a lack of transparency. This project addresses these challenges by designing and implementing a comprehensive Hybrid Recommendation System for Mobiles.

The system integrates Content-Based Filtering, which leverages product specifications, with Collaborative Filtering, which analyzes user behavior, through a sophisticated weighted hybridization strategy. This approach balances deep personalization with the discovery of popular trends. To further enhance user experience and trust, the system incorporates several innovative modules. An **Explainability module** uses LLM-based justification to provide clear, human-readable reasons for its suggestions. Furthermore, to combat filter bubbles and promote discovery, the system introduces **Serendipity** through a feature diversity logic and **Contrarian** recommendations via inverse preference modeling.

The entire system is developed using Python with libraries such as Pandas, Scikit-learn, and Surprise, and is deployed as an interactive web application using Streamlit. The findings indicate that the hybrid model delivers highly relevant and personalized suggestions, while the advanced logic for serendipity and contrarianism significantly increases result diversity and user engagement. The project successfully demonstrates a robust, transparent, and user-centric recommendation engine suitable for real-time deployment in complex choice environments.

Introduction

In the last decade, the global smartphone market has witnessed exponential growth, not just in user base but in the sheer volume and variety of products available. Each month, multiple manufacturers release new models with incremental changes in specifications, features, and pricing. This has flooded the market, transforming a simple purchase decision into a complex research task for the average consumer. Users are faced with a paradox of choice, where an abundance of options leads to decision-making fatigue and confusion.

Recommender systems have emerged as a critical tool to navigate this complexity. By analyzing user preferences and product features, these systems can filter through thousands of options to present a small, manageable set of relevant choices. The two most common approaches are Content-Based (CB) Filtering and Collaborative Filtering (CF).

- **Content-Based Filtering** recommends items similar to those a user has previously liked. For mobiles, this means suggesting a phone with 8GB RAM if the user has shown interest in other 8GB RAM phones. While effective for providing tailored suggestions, this method suffers from over-specialization and rarely introduces novel items.
- **Collaborative Filtering** operates on the principle of homophily—people who agreed in the past will likely agree in the future. It identifies users with similar tastes and recommends items that these "neighbors" have liked, uncovering popular trends and enabling unexpected discoveries. However, it suffers from the "cold-start" problem, where it cannot make recommendations for new users or new items with no interaction data.

Recognizing the key limitations of these individual models, this project proposes a **Hybrid Recommendation System for Mobiles**. By combining the strengths of both CB and CF, the hybrid model aims to solve major issues like the cold-start problem and over-specialization. It balances the user's explicit, feature-based preferences with the implicit, behavior-based wisdom of the crowd, resulting in more robust, accurate, and satisfying recommendations.

Furthermore, this project extends beyond traditional hybrid models by integrating modules for **Explainability, Serendipity, and Contrarian Logic**, creating a truly next-generation recommender system focused on user trust and discovery.

Problem Statement

The primary goal of this project is to design, develop, and deploy a comprehensive hybrid recommender system specifically for mobile phones to address the limitations of existing recommendation models.

The core problem can be broken down into the following key objectives:

1. **Combine Content-Based and Collaborative Filtering:** To develop a hybrid engine that synergizes the strengths of both methodologies, providing recommendations that are both personally relevant and popularly validated.
2. **Generate Personalized and Dynamic Suggestions:** The system must be able to take real-time user preferences (e.g., desired price, camera quality, battery life) and generate a ranked list of tailored phone suggestions.
3. **Enhance Transparency through Explainability:** To combat the "black box" nature of many recommender systems, the project will incorporate a prompt-based explainability module that provides users with clear, concise reasons for each recommendation, thereby fostering user trust.
4. **Introduce Novelty and Discovery:** To prevent filter bubbles and over-specialization, the system will add logic for:
    - **Serendipity:** To suggest relevant but unexpected "hidden gem" items.
    - **Contrarianism:** To challenge user biases by suggesting high-quality alternatives that deviate from their explicit preferences.
5. **Ensure Practical Deployability:** The final system must be packaged into an interactive and user-friendly interface using Streamlit, making it suitable for real-time deployment and demonstration.

This project aims to fill a significant research gap where existing systems often ignore user trust, rarely implement advanced discovery features beyond theoretical models, and lack a specific focus on the nuances of the mobile phone domain.

Literature Review

The literature review explores various hybrid recommendation systems and their components:

- Chandrahaas et al. (2023) proposed a hybrid approach using content-based and collaborative filtering for mobile phone recommendations.
- Biswas and Liu (2022) developed a hybrid recommender system for smartphones, emphasizing personalization.
- Caro-Martínez et al. (2021) introduced an ontological model for explainable recommender systems.
- Tokutake and Okamoto (2023) focused on serendipity-oriented systems using dynamic unexpectedness prediction.
- Smets et al. (2022) discussed serendipity beyond algorithms, emphasizing interface design and user experience.
- Hybrid models using TF-IDF, ALS, and DNN have shown improved prediction accuracy.
- Ontology-based frameworks enhance explainability and user trust.
- Cold-start problems are addressed using deep learning and hybrid techniques.
- Serendipity is influenced by UI design and content diversity.
- Contrarian logic promotes discovery by challenging user biases.
- A thorough review of existing literature was conducted to understand the state-of-the-art in recommender systems, particularly in the context of hybrid models, explainability, and serendipity.
- Several key papers formed the foundation of this project. Chandrahaas et al. proposed a hybrid approach for mobile phone recommendation using Content-Based (TF-IDF, VSM) and Collaborative Filtering. Their work highlighted the suitability of hybrid models for the complex mobile phone market but lacked advanced features like explainability and focused only on standard accuracy metrics (MSE, MAE). Similarly, Biswas and Liu combined Alternating Least Squares (ALS) with a Deep Neural Network (DNN) to address the cold-start problem using side information. While powerful, their abstract lacked specific quantitative results, making it difficult to assess the claimed performance improvements.
- The domain of **Explainable AI (XAI)** in recommender systems is critical for building user trust. Caro-Martínez et al. presented a conceptual model for explainable recommender systems, formalizing it as an ontology to guide development. This work emphasizes that explainability should be a core design requirement, not a simple add-on feature, a principle adopted in this project through the LLM-based justification module.
- The concept of **Serendipity** aims to move beyond mere accuracy to enhance user experience. Tokutake and Okamoto developed a serendipity-oriented system with dynamic unexpectedness prediction, using time-series analysis to boost novelty. Further, Smets et al. argued that serendipity extends "beyond the algorithm" and is heavily influenced by user interface design and content presentation. Their work proposes a feature repository for

designing and evaluating serendipitous systems, highlighting the importance of a holistic, user-centric approach.

- **Synthesis of Literature and Research Gap:**
  The literature confirms that while hybrid models are effective, there is a clear research gap in creating a system that holistically integrates advanced user-centric features. Most existing systems for mobile phones focus purely on accuracy metrics. They often ignore:
  - **Explainability and User Trust:** Users are given recommendations with no justification.
  - **Practical Implementation of Discovery:** Serendipity and contrarian logic are rarely implemented beyond theoretical models.
  - **Mobile-Specific Focus:** General-purpose recommenders fail to capture the unique feature trade-offs inherent in the smartphone market.
- This project directly addresses this gap by creating a single, unified system that combines a robust hybrid filtering engine with dedicated modules for explainability, serendipity, and contrarian logic, all deployed within a practical, interactive interface.

Methodology

The methodology integrates multiple recommendation techniques to overcome individual limitations:

1. Content-Based Filtering: Uses cosine similarity and TF-IDF to match user preferences with product features.
2. Collaborative Filtering: Employs matrix factorization and nearest neighbor algorithms to identify similar users.
3. Hybridization: Combines CBF and CF using a weighted approach (CBF: 70–80%, CF: 20–30%) to balance personalization and popularity.
4. Explainability: Uses prompt-based logic and LLMs to provide human-readable justifications for recommendations.
5. Serendipity: Introduces unexpected but relevant suggestions using feature diversity and time-series reranking.
6. Contrarian Logic: Applies inverse preference modeling to challenge user biases and avoid filter bubbles.

This hybrid methodology is superior to standalone models as it addresses cold-start issues, enhances transparency, and improves user engagement through novelty and diversity.

The methodology for this project is structured around a multi-stage pipeline, from data handling to advanced recommendation modeling and deployment.

4.1. Data Acquisition and Preprocessing

Data Source:
The primary dataset was sourced from Kaggle, titled "Real World Smartphone's Dataset – Gigasheet." This dataset is a merged compilation providing comprehensive information across several categories, including:

- General Product Info & Display
- Performance & Hardware (Processor, RAM)
- Battery & Charging
- Connectivity
- Camera

Challenges and Solutions:
A major challenge was the absence of user interaction data (views, likes, purchases), which is essential for collaborative filtering. While an attempt was made to acquire this via the 91mobiles API, access was denied. For this project, a synthetic user interaction dataset was generated to

simulate real-world user behavior and enable the development of the collaborative filtering module.

Data Preprocessing Pipeline:
A rigorous preprocessing pipeline was implemented to ensure data quality and consistency:

1. **Outlier Removal:** The Interquartile Range (IQR) method was applied to the price column to remove extreme outliers that could skew analysis and model performance.
2. **Handling Missing Values:** Missing numerical values (e.g., processor_speed) were imputed using the **median** of their respective columns, while missing categorical values were filled with the **mode**.
3. **Categorical Variable Encoding:** To prepare the data for machine learning models, categorical variables were encoded. Binary features (2 unique values) were label-encoded to 0/1. Features with few categories (≤10) were one-hot encoded using pd.get_dummies. High-cardinality features were label-encoded.
4. **Data Type Conversion:** All numerical columns were converted to a consistent numeric data type.
5. **Normalization:** Model names and other textual features were cleaned and normalized to handle inconsistencies.

4.2. Exploratory Data Analysis (EDA)

EDA was performed to uncover patterns in the dataset. The price distribution histogram revealed that the majority of phones in the dataset are concentrated in the ₹10,000 - ₹30,000 range. The scatter plot of RAM vs. Price showed a positive correlation but also a high variance, indicating that RAM is not the sole determinant of price.

4.3. System Architecture and Modeling

The system is designed with a modular architecture, where each component performs a specific task before its output is integrated into the final recommendation.

1. Content-Based Filtering (CBF):

- **Approach:** This module uses **Cosine Similarity**. It creates a vector representation for each phone based on its numerical features. When a user provides their preferences, a "user vector" is created. The cosine similarity is then calculated between the user vector and every phone vector.
- **Justification:** Cosine similarity is efficient and effective at measuring the "likeness" between items in a high-dimensional space, making it ideal for matching feature-rich

products like mobiles. It works well even for new users, as it only requires their stated preferences.

2. Collaborative Filtering (CF):

- **Approach:** This module employs user-based **Nearest Neighbor** techniques and **Matrix Factorization**. It analyzes the user-item interaction matrix to find users with similar interaction histories ("neighbors") and recommends items that those neighbors have positively rated but the target user has not yet seen. The Surprise library, a popular Python scikit for recommender systems, was used for this implementation.
- **Justification:** CF excels at capturing trends, popularity, and discovering novel items that a user might not have found otherwise. It complements the limitations of CBF by not being confined to feature similarity.

3. Hybrid Recommender:

- **Strategy:** A **weighted hybridization** strategy was chosen to combine the outputs of the CBF and CF modules. A final score for each item is calculated as:
  Score_hybrid = α * Score_CB + (1 - α) * Score_CF
- **Justification:** A weighted approach is flexible and intuitive. In this project, a higher weight was given to the Content-Based score (α = 0.7-0.8). This is a strategic choice to mitigate the cold-start problem (by relying more on explicit preferences for new users) and to ensure that the recommendations are strongly anchored to the user's immediate needs.

4. Advanced Recommendation Modules:

- **Explainability - LLM-based Justification:**
  - **Approach:** After a recommendation is generated, its key features are fed into a Large Language Model (LLM) using a structured prompt. The prompt asks the LLM to generate a concise, human-readable sentence explaining *why* the phone is a good match based on the user's preferences (e.g., "This phone is recommended because its 5000mAh battery aligns with your preference for long battery life.").
  - **Justification:** This approach provides a high degree of transparency and helps build user trust, moving the system from a "black box" to an interactive advisor.
- **Serendipity - Feature Diversity Logic:**
  - **Approach:** To find "hidden gems," this module introduces a logic that boosts the score of items that are high-quality (high hybrid score) but also less popular or have a diverse set of features compared to the user's typical choices. This involves calculating an "unpopularity" score and using it to re-rank the top candidates.

    - **Justification:** This directly combats over-specialization and introduces novelty, increasing the chance of delightful discoveries and long-term user engagement.
- **Contrarian - Inverse Preference Modeling:**
    - **Approach:** This module identifies items that deviate significantly from the user's explicit preferences but are still objectively high-quality. It calculates a "deviation score" (e.g., using Euclidean distance or squared difference) between the user's preference vector and each item's feature vector. The item with the highest deviation, while still meeting a baseline quality threshold, is presented.
    - **Justification:** Contrarian logic is designed to break the user out of their filter bubble. By presenting a compelling alternative, it encourages exploration and can help users refine their own preferences.

4.4. Deployment

The entire system is deployed as an interactive web application using **Streamlit**. Streamlit was chosen for its rapid development cycle, seamless integration with data science libraries like Pandas and Scikit-learn, and its ability to create an intuitive graphical user interface (GUI) with minimal code.

Modules

The project is broken down into seven distinct software modules, each with a clear responsibility:

1. **Data Handler Module:**
   - **Function:** Loads the mobile phone datasets and user interaction logs.
   - **Responsibilities:** Implements the entire data preprocessing pipeline, including cleaning, handling missing values, encoding categorical data, and formatting the data into structures required by the filtering modules.
2. **Content-Based Filtering Module:**
   - **Function:** Generates recommendations based on item feature similarity.
   - **Responsibilities:** Takes user preferences from the UI, creates a user feature vector, calculates cosine similarity against all items, and outputs a relevance score for each phone.
3. **Collaborative Filtering Module:**
   - **Function:** Generates recommendations based on similar user behavior.
   - **Responsibilities:** Analyzes the user-item interaction matrix, identifies "neighbor" users using algorithms like K-Nearest Neighbors or Matrix Factorization, and generates a list of recommended items.
4. **Hybrid Recommender Module:**
   - **Function:** Combines the outputs of the CB and CF modules.
   - **Responsibilities:** Implements the weighted hybridization strategy to compute a final, unified score for each recommendation. It also performs re-ranking to enhance diversity.
5. **Explainability Module:**
   - **Function:** Provides human-readable justifications for recommendations.
   - **Responsibilities:** Interfaces with an LLM, constructs prompts based on item features and user preferences, and returns the generated explanation text.
6. **Serendipity & Contrarian Logic Module:**
   - **Function:** Introduces novelty and challenges user biases.

- **Responsibilities:** Implements the feature diversity and inverse preference modeling logic to generate unexpected but relevant suggestions.

7. **User Interface Module (GUI):**
    - **Function:** Serves as the user's entry point to the system.
    - **Responsibilities:** Built with Streamlit, it collects user preferences via interactive sliders and widgets, triggers the recommendation pipeline, and displays the final personalized suggestions and explanations in a clear and organized manner.

Results

The evaluation of the system focused on both the quality of the recommendations and the effectiveness of the advanced modules.

- **High Match Rate with User Preferences:** The core hybrid model demonstrated a high success rate in matching recommendations to the user's explicit preferences provided through the sliders. The dominant weight of the content-based component ensured that the primary suggestions were always highly relevant to the user's immediate needs.
- **Increased User Engagement through Serendipity:** In qualitative testing, the serendipitous "hidden gem" picks were found to significantly increase user engagement. Users expressed surprise and interest in devices they had not previously considered but which met their underlying needs, confirming that this feature successfully promotes discovery.
- **Added Diversity from Contrarian Logic:** The contrarian recommendations effectively broke the pattern of highly similar suggestions. By presenting a viable "challenger" option, the system encouraged users to think about feature trade-offs they might have initially ignored, adding valuable diversity to the results.
- **Improved Transparency and Trust:** The prompt-based explanations were a key finding. Users reported a much higher level of trust in the recommendations when they were accompanied by a simple, logical reason. This confirmed that explainability is a crucial component for the adoption of recommender systems.
- **Efficient Real-time Performance:** The system, deployed via Streamlit, demonstrated efficient real-time performance. The combination of optimized algorithms like cosine similarity and the lightweight nature of the deployment framework allowed for recommendations to be generated within seconds, making it suitable for a live, interactive user experience.

Conclusion

This project successfully designed and implemented a sophisticated Hybrid Recommendation System for Mobiles that addresses the critical limitations of traditional models. By integrating Content-Based and Collaborative Filtering, the system delivers highly personalized and accurate suggestions that are robust to the cold-start problem.

The key contributions of this work lie in its extension beyond standard hybrid models. The integration of dedicated modules for **Explainability, Serendipity, and Contrarian Logic** elevates the system from a simple filtering tool to an intelligent discovery platform. Explainability boosts user trust and clarity, serendipity adds novelty and fights recommendation fatigue, and contrarian logic ensures balanced diversity by breaking users out of potential filter bubbles.

The final Streamlit application demonstrates that this multi-faceted approach is not only theoretically sound but also practically feasible for real-time deployment. The project confirms that the future of recommender systems lies in creating holistic, user-centric experiences that prioritize transparency, discovery, and user trust alongside accuracy.

Future Work:

- Integrate real-world, large-scale user interaction data.
- Experiment with more advanced deep learning models (e.g., Neural Collaborative Filtering) for the CF component.
- Conduct A/B testing with real users to quantitatively measure the impact of serendipitous and contrarian recommendations on user satisfaction and conversion rates.
- Expand the explainability module to include visual explanations, such as feature comparison charts.

Acknowledgements

We would like to extend our sincere gratitude to the following:

- **Kaggle**, for providing the comprehensive and high-quality datasets that formed the backbone of this project.
- The global **open-source community**, especially the contributors and maintainers of libraries such as Streamlit, Pandas, Scikit-learn, and Surprise, whose tools were indispensable for this work.
- Our **mentors and project guides** at LTIMindtree and L&T EduTech, for their invaluable guidance, insightful feedback, and constant support throughout the project lifecycle.

References

[1] B. V. Chandrahaas et al., “A Hybrid Approach for Mobile Phone Recommendation using Content-Based and Collaborative Filtering,” EAI Endorsed Transactions on Internet of Things, vol. 10, pp. 1–6, Dec. 2023.

[2] P. K. Biswas and S. Liu, “A Hybrid Recommender System for Recommending Smartphones to Prospective Customers,” arXiv:2105.12876v2, 2022.

[3] M. Caro-Martínez et al., “Conceptual Modeling of Explainable Recommender Systems,” Journal of Artificial Intelligence Research, vol. 71, pp. 557–589, Jul. 2021.

[4] Y. Tokutake and K. Okamoto, “Serendipity-Oriented Recommender System,” Proc. IEEE SMC, Hawaii, USA, Oct. 2023.

[5] A. Smets et al., “Serendipity in Recommender Systems Beyond the Algorithm,” IntRS’22 Workshop, CEUR Workshop Proceedings, vol. 3222, 2022.