IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 11, November 2026

AN AI-DRIVEN PERSONALIZED PRODUCT RECOMMENDATION FRAMEWORK USING USER–ITEM INTERACTIONS

Urvashi[1], Tejaswini[2], Sohil[3]

Department of Computer Science and Engineering (Artificial Intelligence and Machine Learning),

SRM Institute of Science and Technology, Vadapalani Campus, Chennai, India[1-3]

Abstract: The rapid growth of e-commerce platforms has made it increasingly difficult for users to identify products that match their individual preferences. Traditional product browsing methods often present a large number of choices without considering the unique interests and behaviour of each user. This paper presents a proposed AI-driven personalised product recommendation framework that analyses user preferences, product characteristics, and user–item interactions. The framework considers content-based filtering and collaborative filtering to identify relevant products and generate ranked recommendations. Its processing stages include data collection, preprocessing, user and product profiling, recommendation scoring, and ranking. Evaluation measures such as Precision@K, Recall@K, F1-score, Mean Absolute Error, and Root Mean Squared Error are considered for assessing recommendation quality, where applicable. The framework aims to reduce information overload and improve product discovery in e-commerce environments. Experimental results are not reported in this draft and must be added after implementation and evaluation using a verified dataset.

Keywords: Personalized Recommendation System, Artificial Intelligence, Machine Learning, Product Recommendation, Content-Based Filtering, Collaborative Filtering, E-commerce, User–Item Interaction.

I. INTRODUCTION

The increasing availability of products through online shopping platforms has exposed users to a large volume of choices. Although a wide product selection provides convenience, finding suitable products can be time-consuming. Users have different interests, budgets, preferences, and purchasing behaviours, making identical product displays less effective for individual needs.

A recommender system is an information-filtering system that predicts the relevance of items to a particular user and presents suggestions based on available information. Machine learning enables recommendation systems to identify patterns in user–item interactions such as ratings, clicks, views, and purchases.

Common recommendation techniques include content-based filtering and collaborative filtering. Content-based filtering uses product attributes and user interests, whereas collaborative filtering identifies similarities in the behaviour of users or interactions with items. Hybrid methods combine multiple techniques to address some of their individual limitations [1].

A. Problem Statement

Users of online shopping platforms may find it difficult to identify products matching their preferences because of the large number of available choices. A recommendation mechanism that does not adequately consider user behaviour may produce irrelevant or repetitive suggestions. A personalised framework is therefore proposed to analyse user–item interactions and product characteristics.

B. Objectives

- To collect and preprocess user–item interaction data.
- To create user profiles from available preferences and interaction history.
- To represent products using relevant product attributes.
- To investigate content-based and collaborative filtering techniques.
- To generate and rank recommendations according to predicted relevance.
- To evaluate recommendation quality using appropriate metrics.

© IARJSET This work is licensed under a Creative Commons Attribution 4.0 International License

II. RELATED WORK

Adomavicius and Tuzhilin [1] presented an overview of recommender system techniques and classified recommendation approaches into content-based, collaborative, and hybrid methods. Their work discussed limitations of existing techniques and the need for more comprehensive personalisation.

Content-based filtering recommends products by comparing their characteristics with a user's interests. It can generate recommendations when information about other users is unavailable, but may produce suggestions that are too similar to items with which the user has already interacted.

Collaborative filtering uses preferences or interaction patterns from multiple users to identify products that may interest a target user. Its effectiveness depends on the availability and quality of interaction data. Sparse interaction matrices and new-user or new-item situations can create challenges.

Herlocker et al. [2] examined the evaluation of collaborative filtering recommender systems and highlighted the importance of selecting appropriate datasets, evaluation tasks, and prediction-quality measures.

A. Research Motivation

Content-based methods depend heavily on product attributes, while collaborative filtering requires sufficient interaction information to identify meaningful patterns. This work is motivated by the need to organise these approaches into a modular product recommendation framework. Its effectiveness must be established through implementation and experimental comparison.

III. PROPOSED METHODOLOGY

The proposed framework follows a sequence of processing stages, beginning with data collection and ending with ranked product recommendations.

A. System Architecture

The system is organised into six modules: (1) user data collection, (2) data preprocessing, (3) user and item profiling, (4) recommendation engine, (5) prediction and ranking, and (6) evaluation.

User and Product Data → Preprocessing → User and Item Profiling → Recommendation Engine → Scoring and Ranking → Personalised Recommendations

Fig. 1. Proposed workflow of the product recommendation framework.

B. Data Collection and Preprocessing

The framework requires product information such as product identifier, name, category, description, brand, and other available attributes, together with user interaction information such as user identifier, product identifier, rating, view, click, or purchase interaction, depending on dataset availability.

Preprocessing may include removing duplicate records, handling missing or inconsistent values, normalising text attributes, encoding categorical variables, converting user–item interactions into a suitable representation, and separating training and testing data. The final procedure depends on the dataset used.

C. Content-Based Filtering

Content-based filtering recommends products by measuring similarity between product characteristics and a user's interests. Textual product attributes can be converted into numerical representations using Term Frequency–Inverse Document Frequency (TF-IDF). The inverse document frequency component can be expressed as:

$$IDF(t) = \log(N / df(t))$$

where N is the total number of product documents and df(t) is the number of product documents containing term t. Cosine similarity can measure similarity between two product vectors:

$$Cosine(A,B) = (A \cdot B) / (||A|| \, ||B||)$$

A higher cosine similarity indicates greater similarity between represented product attributes. A user profile may be constructed from products with which the user has interacted, and candidate products may be compared against that profile.

D. Collaborative Filtering

Collaborative filtering identifies recommendation patterns using interactions among users and products. A user–item interaction matrix is represented as R = [r_ui], where r_ui represents the interaction or rating of user u with product i. User-based collaborative filtering identifies users with similar interaction patterns, while item-based collaborative filtering identifies products with similar interaction patterns across users. Performance may be affected by sparse interaction data.

E. Hybrid Recommendation Approach

A hybrid approach may combine content-based and collaborative filtering scores using a weighted formulation:

$$S(u,i) = \alpha\, S_CB(u,i) + (1 - \alpha)\, S_CF(u,i)$$

Here, S(u,i) is the combined score for user u and product i; S_CB and S_CF are the content-based and collaborative filtering scores; and α is a weighting parameter between 0 and 1. Component scores must be placed on comparable scales, and α should be selected through validation. This is a proposed formulation; implementation must be confirmed.

F. Recommendation Ranking

Candidate products are sorted in descending order of their recommendation scores. The system can return the top-K products for a user. Filters or exclusion of previously interacted products may be applied depending on the final implementation.

IV. SYSTEM REQUIREMENTS

The proposed software environment includes Python 3.x, Jupyter Notebook or Google Colab, Pandas, NumPy, Scikit-learn, and Matplotlib. Flask or FastAPI may be used for backend integration, while HTML, CSS, JavaScript, or React may be used for the user interface. MySQL or MongoDB may be used for storage if required. The final manuscript must list only technologies actually used.

The framework can be developed using a standard laptop or desktop computer with a modern multi-core processor, at least 8 GB RAM recommended, sufficient storage, and an internet connection where online resources or cloud environments are used. A dedicated GPU is not essential for basic recommendation techniques.

V. EVALUATION METHODOLOGY

Recommendation systems should be evaluated using metrics that measure the quality of recommended products and, where applicable, the accuracy of predicted ratings.

A. Precision@K and Recall@K

Precision@K measures the proportion of recommended products among the top K results that are relevant: Precision@K = (Relevant products in top K) / K. Recall@K measures the proportion of all relevant products that appear in the top K recommendations: Recall@K = (Relevant products in top K) / (Total relevant products).

B. F1@K

F1@K combines precision and recall using their harmonic mean: F1@K = 2 × (Precision@K × Recall@K) / (Precision@K + Recall@K).

C. MAE and RMSE

When numerical ratings are predicted, Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) can measure prediction error. $MAE = (1/n)\, \Sigma\, |r_i - \hat{r}_i|$. $RMSE = \sqrt{[(1/n)\, \Sigma\, (r_i - \hat{r}_i)^2]}$, where r_i is the actual rating and $\hat{r}_i$ is the predicted rating. These measures are appropriate only when numerical rating predictions are produced.

VI. RESULTS AND DISCUSSION

The project planning material available for this manuscript does not contain verified experimental results. This section must be completed after the system has been implemented and tested. No performance values are claimed in this draft.

The final evaluation should report the dataset source and size, number of users, products and interactions, preprocessing details, implemented algorithms, train–test strategy, measured ranking metrics, and actual recommendation examples.

TABLE I. DATASET CHARACTERISTICS

Parameter	Value
Number of users	To be measured
Number of products	To be measured
Number of interactions	To be measured
Dataset source	To be specified

TABLE II. EXPERIMENTAL PERFORMANCE (TO BE COMPLETED AFTER TESTING)

Metric	Content-based	Collaborative	Hybrid
Precision@K	Pending	Pending	Pending
Recall@K	Pending	Pending	Pending
F1@K	Pending	Pending	Pending

VII. CONCLUSION

This paper presents an AI-driven personalised product recommendation framework designed to generate product suggestions using user–item interaction data and product characteristics. The proposed architecture includes data collection, preprocessing, user and item profiling, recommendation generation, ranking, and evaluation. Content-based filtering and collaborative filtering are considered as recommendation approaches, while a hybrid strategy is proposed to combine their respective strengths. The final effectiveness of the system must be established through implementation and experimental testing using verified data. Future work may include improving cold-start handling, incorporating contextual information, using advanced machine learning techniques, and evaluating the system with a larger and more diverse product dataset.

REFERENCES

[1]. G. Adomavicius and A. Tuzhilin, “Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions,” IEEE Transactions on Knowledge and Data Engineering, vol. 17, no. 6, pp. 734–749, 2005, doi: 10.1109/TKDE.2005.99.

[2]. J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl, “Evaluating Collaborative Filtering Recommender Systems,” ACM Transactions on Information Systems, vol. 22, no. 1, pp. 5–53, 2004, doi: 10.1145/963770.963772.

[3]. P. Lops, M. de Gemmis, and G. Semeraro, “Content-based Recommender Systems: State of the Art and Trends,” in Recommender Systems Handbook, F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, Eds. Springer, 2011, pp. 73–105.