International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(20s) 2026 pp. 184-204
https://doi.org/10.70917/ijcisim-2026-5164
Article

AI-Driven Personalized Shopping Recommendation Framework Using Small Language Models and User Behavior Analytics

Harshita Chaurasiya[1], Aditi Wangikar[2], Swati Mugale[3], Ravi Ray Chaudhari4*, Sarika Jadhav[5], Anurag Rai[6], Manvi Chopra[7]

1,4 Amity School of Engineering and Technology, Amity University Madhya Pradesh, Gwalior, India
[2]Department of Computer Engineering, Marathwada Mitra Mandal's college of Engineering, Karve Nagar, Pune, India
[3]Department of AI&DS, Dr. D. Y. Patil Institute of Technology, Pimpri, Pune, India
[5]Department of CSE - ASET, Keystone School of Engineering, Pune, India
[6]Department of Computer Science & Applications, Sharda School of Computing Science and Engineering, Sharda University, Greater Noida, India
[7]School of Engineering and Computing, Dev Bhoomi Uttarakhand University, Dehradun, India
Email: 1harshitachaurasiya27@gmail.com, 2aditiwangikar@outlook.com, 3mugaleswati12@gmail.com, 4*glaitmravi@gmail.com, 5dr.sarikabodake@gmail.com, 6aav040572@gmail.com, 7chopramanvi4@gmail.com
*Corresponding Authors

Abstract: In today's e-commerce landscape, personalized recommendation is a critical tool for enhancing customer engagement and decision-making. The traditional recommendation methods, however, have the disadvantages of weak user interaction, cold start issue, insufficient contextual information and high computing demand of large language models. This study introduces the concept of an AI-Driven Personalized Shopping Recommendation Framework, combining user behavior analytics and Small Language Models (SLMs) to effectively recommend products based on user context. The framework examines visitors' purchase frequency, browsing habits, search history, dwell time, category preferences, brand affinity, and price sensitivity to build dynamic user profiles. SLMs are used for contextual preference summarization, semantic product understanding and personalized recommendation generation. A hybrid ranking method is used which integrates behavioral, semantic similarity, SLM preference scores and product popularity, and an explainability layer is created to generate recommendation reasons and confidence scores. The experimental results prove that the framework outperforms conventional recommenders and deep learning recommenders in terms of the recommendation quality, as well as the inference latency and computational requirements, thus making it a potential solution for a real-world e-commerce application that needs to be scalable.
Keywords: Small Language Models, Recommendation Systems, Personalized Shopping, User Behavior Analytics, Explainable AI, Retail Analytics, E-commerce, Consumer Modeling

1. Introduction

Ecommerce has revolutionized the retail market, allowing people to obtain millions of products from all over the world via Internet technology. The rise of internet penetration, smartphone usage and the progress of cloud computing has made online shopping more quickly and brought in a ton of information regarding user interactions with the web, including their browsing history, search history, clickstreams, purchasing information, product ratings, and reviews. The abundance of behavioral information is significant and offers a great opportunity to create a system that can assist in building intelligent recommender systems that can provide personalized shopping experiences [1]. For online retailers, the personalised recommendations help them in two ways: they help customers find relevant products, and they enhance customer satisfaction, engagement and loyalty, boost sales and conversion rates.

Artificial Intelligence (AI) has emerged as a fundamental pillar in today's recommendation systems, facilitating a data-driven and adaptive approach to personalization. The traditional recommendation methods such as collaborative filtering and content-based filtering have slowly developed into highly advanced AI-based algorithms, which utilize machine learning, deep learning, knowledge graph, and natural language processing to make accurate recommendations [2]. The ability to grasp the semantic relationships between products, customer preferences, and contextual information has been made possible by recent progress in transformer-based language models, which greatly enhances the functionalities of recommender systems. These models can read product descriptions, customer reviews, and even the conversation to make very personal suggestions [3]. As a result, AI-powered recommendation systems have become essential tools in the e-commerce industry, assisting shoppers in navigating through vast product catalogues and enhancing business outcomes by delivering personalized recommendations.

The recommendation data system is the one of the major solutions to the challenge of data deluge in online retail settings [4]. When product stock continues to grow, customers frequently have trouble finding products that fit their taste, price and needs. Relevant product recommendations based on interest and patterns of behavior cut down search effort, boost user satisfaction, drive click-throughs and help retain customers. What's more, personalization helps generate revenue as it boosts the average order value and repeat purchase rates and reduces customer churn. Hence, designing intelligent, scalable and efficient recommendation frameworks has emerged as a critical research challenge in academia and industry [5].

Although there has been significant progress in the field of recommender systems, there are several issues that plague existing systems in the context of dynamic e-commerce settings [6]. One of the most popular recommendation methods, collaborative filtering greatly depends on user-item interaction matrices to forecast customer preferences. It has, however, been found that its performance is poorer when there are a lack of interaction data or new users and products are introduced in the system. This is a common cold-start issue and hinders the ability to generate accurate recommendations for customers who have not made many purchases and new products that have not generated many interactions. Another drawback is the lack of scalability. Recommendation algorithms must be able to deal with many users and products to handle today's systems that process millions of users and products simultaneously without consuming too much computational power. With the growth of datasets, classical collaborative filtering and deep learning recommendation models tend to become more complex and slower, which challenges their ability to make real-time recommendations.

Large Language Models (LLMs) have brought powerful semantic reasoning powers to the recommendation systems. Transformer-based models like GPT have shown significant capabilities in comprehending user intent and context, as well as product semantics. However, such models need billions of parameters, high computational power, costly hardware accelerators, and high energy usage during training and inference. This computational property makes them too impractical to be used in many real-world recommendation applications, including on the edge devices and resource-constrained environments [7]. Another critical challenge is privacy preservation. Personalized recommendation systems gather highly personal data from users, such as their browsing habits, shopping history, demographic information, place of residence, and preferences for shopping. These data can lead to privacy issues and unauthorized profiling if they are handled improperly. Moreover, with the ever-tightening regulatory standards around personal data protection, recommendation models need to provide accurate personalization with as little data exposure and complexity as possible. Thus, there is a need for recommender systems that have high recommendation accuracy, computational efficiency, scalability and privacy preservation.

The shortcomings of the previous recommendation methods push the development of lightweight intelligent recommendation framework, which can provide high-quality recommendation of personalization while having lower computation costs. In recent years, SLMs with significantly fewer parameters have been shown to perform competitively on a range of natural language understanding tasks with reduced memory use, faster inference and lower deployment costs. The above features make SLMs extremely valuable for real-time recommendation systems that require efficient use of resources and require near real-time processing. Unlike traditional recommendation systems which mostly look at previous interactions, SLMs can understand the semantic relationships among user preferences, product details, reviews, and context [8]. SLMs can be used in combination with user behavior analytics to create a more comprehensive user representation based on browsing patterns, purchasing frequency, search behavior, shopping session duration, product categories, shopping intention related to time, and contextual shopping intentions. The hybrid learning approach is used to achieve more accurate personalization with less reliance on a long list of interactions [9].

The other key driving factor is the growing need to generate recommendations in real time. Today's customers want targeted recommendations when they're shopping online. High inference latency can have a negative impact on the user experience and result in lower conversion rates. Compared to Large Language Models, Small Language Models consume less computational resources, making them faster, more efficient, and more deployable on cloud edge mobile platforms [10]. Therefore, it is a plausible way of designing effective, scalable, explainable, and privacy-preserving recommendation systems for the future of e-commerce, by combining user behaviour analysis and SLMs.

The goal of this study is to create a personalized shopping recommendation system based on AI technology that integrates the SLM and user behavior analysis to provide accurate, context-aware and computationally efficient product recommendations. To accomplish this goal, the study pursues following specific objectives:

- To develop a lightweight and intelligent Shopping Recommendation System based on AI technologies combining semantic understanding with behavioral intelligence for personalized product recommendation.
- To create a robust user behavior analytics module that is able to provide significant features from browsing history, transaction history, search history, clickstream data, ratings, and customer engagements.
- Use SLM to achieve context-based understanding of user preferences and product description and dramatically outperform LLMs in terms of computational complexity.
- Build a hybrid recommendation system that would benefit from using a combination of behavioral features and semantic representation to enhance the quality of recommendations and tackle data sparseness.
- Improve the performance of recommendations by increasing Precision, Recall, F1-score, Normalized Discounted Cumulative Gain (NDCG), Mean Average Precision (MAP) and customer satisfaction indicators.

This research can be summarized as follows:

A novel AI-driven personalized shopping recommendation framework, which combines Small Language Models and user behavior analysis, is proposed to enable efficient and context-aware product recommendation. A hybrid user behavior modelling approach is designed that combines user browsing behavior, purchasing behavior, search history, session features, category preferences and user interaction to create user profiles. To achieve this, a lightweight module for semantic recommendations is designed using Small Language Models to grasp the meaning of products and customers, significantly cutting down on computational resources, memory usage, and inference time. The development of a dynamic context-aware recommendation strategy that takes into consideration temporal behavior, user interests, shopping sessions and contextual preferences to enhance the relevance of recommendations. There is an explainable recommendation mechanism that is integrated, giving users clear explanations of why a product have been recommended, which enhances the interpretability of AI recommendations and boosts user trust.

The rest of this paper will be structured as follows. In Section 2, an extensive survey of the literature related to personalized recommendation systems, user behavior analytics, transformer-based recommendation models, and Small Language Models is presented, and current research gaps are identified. In Section 3, the proposed AI-powered personalized shopping recommendation framework is detailed, comprising of the overall framework, user behaviour modelling process, integration of Small Language Models, recommendation generation module, and explainability module. The major details of the experimental setup, benchmark datasets, implementation details, baseline methods, evaluation metrics and parameter settings used for performance evaluation are described in Section 4. The experimental results, comparative performance analysis, ablation studies, computational efficiency evaluation, and discussion of the proposed framework is presented in Section 5. Lastly, Section 6 summarizes the main results of the paper, emphasizing the practical relevance of the recommended framework for intelligent and privacy preserving personalized shopping systems, outlines the limitations of the proposed research, and outlines some future research directions.

2. Literature Review

In recent years, personalized recommendation systems have shown great advancements in the use of deep learning, graph-based learning, and transformer-based approaches. In this study, researchers [11] introduced an improved version of the Neural Collaborative Filtering (NCF) framework, which incorporated nonlinear neural networks for learning the latent user–item interactions more effectively than traditional matrix factorization methods. They designed a method that captured complex preferences of users and resulted in significant accuracy gains using benchmark datasets like MovieLens and Pinterest. But the model had still significant reliance on historical interaction data and had cold-start issues and lack of data. Likewise, authors [12] proposed BERT4Rec, a transformer-based sequential recommendation model using bidirectional self-attention to capture the context relationship between the sequence of user interactions. They found their model to be better in terms of considering long-term dependencies than recurrent neural network-based models, but it was very costly in terms of computation and memory usage, which made it hard to implement it in real-time recommendation systems. In recent years, Session aware recommendation has also received a lot of attention. Researchers [13] proposed a framework, GRU4Rec, with Gated Recurrent Units (GRUs) to capture sequential user behaviour within a shopping session. Even with such enhancements, the model proved itself to be hard to capture very long-term user interests and had a performance drop if their session history was very variable. Similarly, by adding temporal shopping patterns, authors [14] explored the LSTM based recommendation system for forecasting future purchases. While their approach was successful in modeling sequential purchasing behaviour, it was very time consuming to train and needed a significant amount of computing power, making it difficult to scale in large commercial platforms.

A new research field that has been gaining popularity is the graph-based recommendation approach. Researchers [15] proposed a recommendation framework based on Graph Neural Network (GNN) which is able to learn the high-order relations among users and products in the large-scale user-product interactions graph. They were able to improve the diversity of recommendations and the accuracy of prediction by leveraging graph neighborhood information. However, the creation and upkeep of large-scale interaction graphs used significant computer resources, making real-time deployment somewhat difficult. Likewise, Xiang Wang et al. suggested recommendation approaches based on graph convolution networks and introduced heterogeneous information networks to capture the complex relationships between users and items. These methods were successful for representation learning, but were expensive in computation and hard to update online as new interactions constantly came. Other models based on attention mechanism have successfully furthered the development of personalized recommendation by dynamically determining user interests. Researchers [16] introduced the Deep Interest Network (DIN) that uses attention mechanisms to assign different weights to the historical user behaviors based on the target products. The experimental results proved that the proposed approach achieved better prediction for CTR and relevance for recommendation in large-scale industrial datasets. The model, however, was mostly based on structured interaction data and had limited ability in comprehending the semantic information found in the textual product descriptions and customer feedback. Authors [17] added knowledge graphs to the recommendation system to boost the semantic reasoning and diversifying the recommendation list. They were able to successfully relate users, products, brands and categories on a graphical level, enhancing explainability. However, the ability of building up extensive knowledge graphs was demanding and domain-specific, limiting the adaptability between heterogeneous eCommerce platforms.

Researchers [18] used CNN architectures to capture the semantically meaningful representation of product reviews and descriptions. Their approach to sentiment-aware recommendation was to detect local textual patterns and user's opinions. The CNNs were able to model the context of textual information well, but were not very effective at capturing long-range dependencies in sequential user interactions. Similarly, customer activity-based temporal modeling has been proposed in sequential recommendation models by authors [19], which showed to better predict future purchases. Although they achieved a higher accuracy in their recommendation results, they consumed a lot of computation power and training time compared to traditional machine learning methods. LLMs have revolutionized recommendation research by bringing cutting-edge semantic understanding and reasoning. Researchers [20] gave a comprehensive survey of the LLM-based recommendation systems, which can be divided into representation learning, conversational recommendation, user profiling and recommendation generation. They discussed the benefits of transformer-based language models in their study, which noted its ability to provide a richer context, but also outlined some limitations such as computational expense, privacy concerns, and deployment delays. Researchers [21] suggested a generative recommendation approach that directly generates personalized recommendations based on natural language reasoning using GPT-based architectures. Their methodology enhanced the contextual personalization and recommendation variety considerably. But hallucination problems, high inference costs and reliance on powerful computing infrastructure were still significant drawbacks.

Given the practical difficulties with the larger models, there have been several recent attempts to examine alternative architectures for providing recommendations. In [22] summarized the adaptation strategies of next-generation LLM-based recommendation systems, retrieval augmentation, prompt engineering and fine-tuning methods. Their study showed that LLMs can effectively interpret user intent in queries, but highlighted the need for better deployment strategies in resource-limited settings. Researchers [23] discussed the applications of LLMs for recommendations, with a focus on zero-shot recommendations, conversation-based recommendations, and explainability. To make the recommendation systems more efficient and scalable, recent investigations have focused on it. In [24] performed a comprehensive review of LLM applications in recommendation systems, which included representative learning, ranking, retrieval and conversational recommendation. Their survey found that although LLM systems show significant improvements in recommendation quality, the problem of getting real-time recommendations with reasonable computation time is still an open question. In the same vein, LLM-powered recommendation agents that can reason, remember, and plan have been proposed by [25]. While their architecture enhanced recommendation intelligence, it added to the orchestration complexity and boosted inference latency, making it costly to deploy for commercial e-commerce platforms.

A few further review studies have confirmed the need to design lightweight recommendation systems. Researchers [26] thoroughly examined the use of LLMs in various recommendation tasks, datasets, architectures, and evaluation methods. They found that previous research was largely focused on LLMs with fewer resources dedicated to SLMs. Researchers [27] made a comparison between prompting, fine-tuning, retrieval augmented generation, and hybrid recommendation strategies. Their analysis revealed that while LLM-powered methods delivered remarkable recommendation results, there is still limited standardization in evaluation frameworks and lightweight deployment solutions. There have been few studies that study the recommendation problem from a problem-oriented view. Researchers [28] systematically reviewed problems in LLM-based recommendation systems, including cold-start, interpretability, efficiency, fairness and user experience. They concluded that there was no existing framework that could meet all the requirements of recommendation quality, computational efficiency, explainability and deployment scalability. In addition [29] proposed multimodal recommendation frameworks combining text-based product descriptions, product images and structured behavioral data. Multimodal fusion led to a further improvement in personalization, but also raised the complexity of the computations and the training of the models.

Recently, new avenues for efficient recommendation systems have become possible with the advent of Small Language Models research. Compact language models like Phi, Gemma, TinyLlama and SmolLM have shown to achieve competitive reasoning capabilities while significantly reducing inference latency, memory usage and deployment costs. These approaches are well suited for real-time recommendation and edge computing. However, existing research focuses mainly on general natural language understanding tasks and has not been extensively investigated for personalized shopping recommendation systems which use extensive user behaviour data, such as clickstream data, search history, purchase frequency, dwell time, cart abandonment, wish list data, ratings and session length. Overall, the literature reviewed shows a clear shift from the traditional collaborative filtering methods to deep learning, graph neural networks, transformers, and Large Language Model-based recommendation systems. These methods have made significant strides in recommendation accuracy, contextual understanding and conversational intelligence, but still require significant amounts of computing power, lack scalability, privacy concerns, aren't easily explainable and deployment is costly. Furthermore, few works explain how to combine SLM models with a thorough user behavior analytics to enable semantic reasoning, computational efficiency, real-time inference, explainability and scalable personalization. To overcome these shortcomings, this research introduces a novel and powerful AI-based personalized shopping recommendation system that integrates multi-dimensional user behavior analysis with SLMs, offering accurate, explainable, resource-efficient and context-sensitive recommendations for contemporary e-commerce platforms.

Most of the traditional recommendation methods still suffer from few interactions data, cold-start problems, not enough contextual information, and scalability problems. Despite their impressive capabilities in terms of semantic reasoning, Large Language Models are still limited by their very large number of parameters, slow inference speed, high memory usage and energy consumption, and high cost of operation. In addition, much of the previous research is mainly on the recommendation accuracy and relatively less attention is given to the computational efficiency, real-time deployment, explainability, and privacy-aware recommendation. While the potential applications of Small Language Models in e-commerce are promising, the combination of user behavior analytics with the lightweight models has not been widely explored. So, there is a clear research gap to design an efficient recommendation framework that can provide contextual understanding, behavioral personalization, computational efficiency, explainability, and real-time recommendation generation. This proposed research seeks to fill this void by combining user behavior analytics and Small Language Models to build an easily explainable, lightweight, and scalable personalized shopping recommendation system.

Table 1. Literature Review

References	Year	Methodology Used	Major Contributions	Research Gap Identified
[15]	2022	GRU4Rec (RNN)	Enhanced session-based recommendation using sequential behavior modeling.	Weak long-term dependency learning.
[16]	2023	Deep Interest Network (DIN)	Adaptive attention over user interests improved click-through rate prediction.	Limited semantic understanding of user intent.
[17]	2023	Knowledge Graph Recommendation	Integrated entity relationships to enhance recommendation diversity and explainability.	Knowledge graph construction is resource intensive.
[18]	2024	Large Language Model-based Recommendation Survey	Established taxonomy of LLM integration into recommender systems and identified major application domains.	Practical deployment challenges including latency, privacy, and cost remain unresolved.
[19]	2024	Generative LLM Recommendation	Proposed generative recommendation that directly predicts relevant items using LLM reasoning.	Hallucination, inference cost, and scalability challenges.
[20]	2024	Next-Generation LLM-based Recommendation	Presented industrial taxonomy covering representation, reasoning, and deployment.	Limited lightweight deployment strategies.
[21]	2024	LLM-enabled Recommendation Review	Highlighted contextual reasoning, zero-shot learning, and explainable recommendation.	Prompt sensitivity and computational expense.

[22]	2025	Comprehensive LLM Survey	Categorized LLM applications into representation, ranking, and conversational recommendation.	Efficient real-time recommendation remains an open problem.
[23]	2025	LLM-powered Recommendation Agents	Introduced agent-based recommendation using planning, memory, and reasoning capabilities.	High computational overhead and complex orchestration.
[24]	2025	Comparative Analysis of LLM Recommenders	Compared prompting, fine-tuning, retrieval, and hybrid LLM recommendation strategies.	Need for standardized benchmarking and efficient deployment.
[25]	2025	Problem-driven LLM Recommendation Survey	Organized LLM recommendation research around cold-start, interpretability, user experience, and efficiency.	Lack of unified solutions balancing performance and resource efficiency.
[26]	2025	Multimodal LLM Recommendation	Combined text, image, and structured information for richer personalization.	Increased computational complexity and multimodal fusion challenges.
[27]	2025	Phi, Gemma, TinyLlama, SmolLM	Demonstrated reduced inference latency, lower memory usage, and competitive reasoning for edge deployment.	Limited evaluation for personalized shopping recommendation and behavioral analytics integration.

3. Proposed AI-Driven Personalized Shopping Recommendation Framework

The proposed AI-based personalized shopping recommendation system combines user behavior analysis with SLMs to provide accurate, context-aware, explainable, and computational efficient product recommendations. In contrast to traditional collaborative filtering based or deep neural network-based recommendation systems, the proposed approach fuses explicit user behavioral features, with implicit ones, and leverages lightweight language models for semantic reasoning. The main goal is to provide very personalized recommendations with minimum computational complexity, inference delay and memory usage.

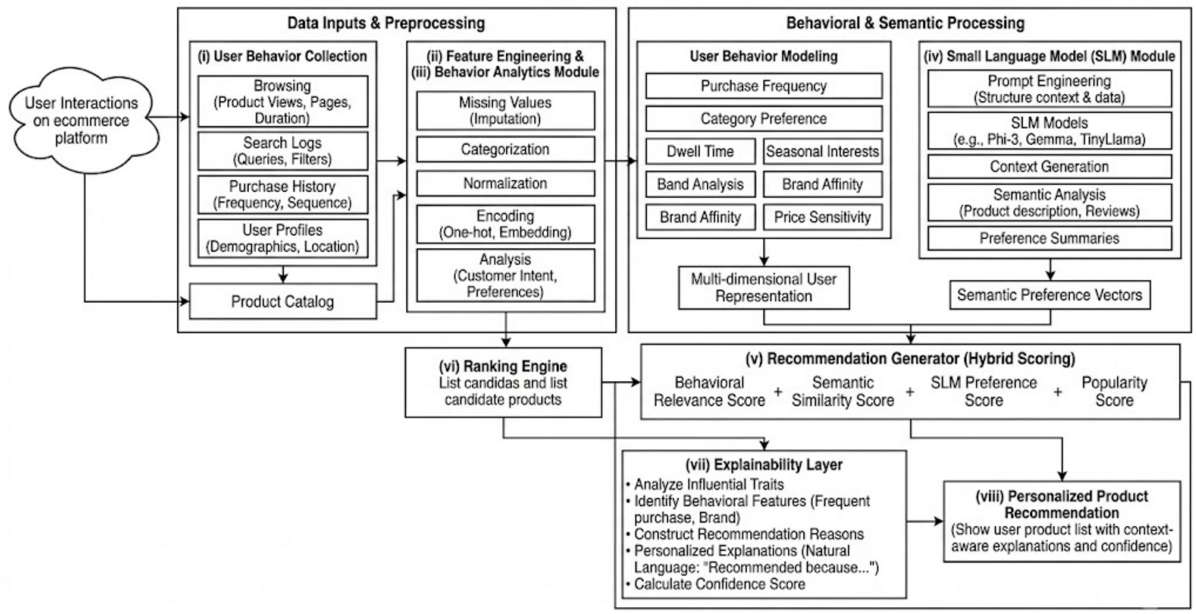

Figure 1: Proposed Personalized Shopping Recommendation System Architecture with Behavioral and Semantic Intelligence

The overall architecture is divided into eight interconnected modules: (i) User Behavior Collection, (ii) Feature Engineering, (iii) Behavior Analytics Module, (iv) Small Language Model Module, (v) Recommendation Generator, (vi) Ranking Engine, (vii) Explainability Module, and (viii) Personalized Product Recommendation. The entire process starts with gathering multidimensional user behavior data as users engage with the ecommerce platform. These data are then processed using feature engineering to convert them into meaningful numerical representation and analyzed to gain insights into customer preferences and shopping intentions. These behavioral attributes are combined with the Small Language Model's semantic representations to form a comprehensive customer profile. The hybrid recommendation engine uses a combination of behavioral relevance, semantic similarity, contextual understanding and product popularity to calculate a recommendation score for the candidate products. Finally, an explainability layer offers clear explanations of the recommendations and their corresponding level of confidence before showing personalized products to end users. The proposed architecture has a few benefits over the traditional recommendation system (Figure 1). Firstly, user behaviors can be analyzed and incorporated into SLMs, allowing a greater understanding of customer intentions beyond just historical purchase data. Second, lightweight models are significantly more efficient at computation than Large Language Models yet retain a comparable contextual reasoning ability. Third, its modular design enables it to be scaled across cloud servers, edge devices, and mobile commerce platforms. Finally, the added explainability module increases the trustworthiness of the generated recommendations by offering interpretable explanations. The overall flow of the proposed framework is shown below in Figure 1 that shows the flow of information from the user interaction to the personalized product recommendation through the behavior analytics and semantic reasoning modules.

Data Collection: A crucial aspect of a high-quality recommendation system is the availability of detailed and varied user data. Thus, the proposed framework integrates heterogeneous data from various sources to comprehensively model customer preferences, product features, and contextual customer shopping behavior. Collected data is segmented in such a way that the information can be divided into two categories: product-oriented information and user-oriented behavioral information. There are product attributes such as product identifiers, product description, technical specification, category label, brand information, availability, product price, seller information, price percentage, customer ratings, and multimedia that are detailed in the product catalog. These structured and unstructured properties enable the semantic understanding of products and comparison of customer interests and set of products.

User profile includes data related to demographics, account, geographical location, language preference, purchasing category, preferred payment, membership, and historical shopping interests. Demographic data can be used to profile users but with the proposed framework behavioral features rather than static user data are used to increase the level of personalization in recommendations. The shopping history provides details of the previously bought products, frequency of purchase, time between purchases, the value of orders, the categories of products purchased, and the purchasing sequence. Past buying habits can be useful in understanding long-term customer preference and customer buying habits.

The behavior of customers when they visit a website is captured implicitly through browsing behavior, such as product views, pages visited, browsing sequence, duration of browsing and scrolling behavior. These interactions can often expose customer interests while waiting for their purchases and are a key factor in personalized recommendation. The framework also includes search logs including user-given search queries, keyword frequencies, product filters, and search refinement patterns. Search history can be used to identify explicit customer intent and help to understand customers' current shopping goals. Furthermore, the product ratings give clear information about customer satisfaction, and customer reviews have a rich amount of textual information that documents the opinions, product quality, benefits, and drawbacks of customers. Structured datasets have no semantic information in the reviews, which is crucial for SLM. These disparate data sources combined can give a complete description of customer preference, and the proposed recommendation system can be able to reflect both explicit and implicit behavioral indicators.

Data Preprocessing: Raw behavioral and product data from e-commerce environments often includes inconsistencies, redundancies, gaps, textual noise, and a variety of data formats. Therefore, it is critical to have an efficient preprocessing phase to enhance data quality and recommendation results. The first step is to deal with missing values. On numerical attributes like product ratings, frequency of purchases, session duration etc., mean or median imputation is performed based on the statistical distribution of the attribute. Other categorical data such as product category and brand information are coded as the most common category or as unknown categories to retain important information. After cleaning the data, it is then normalized to remove variations in feature scales. The behavioral attributes (e.g. purchase frequency, session time, product prices) have different numerical ranges, so Min-Max normalization is used to scale all continuous features to the range [0, 1] to ensure that features with a higher numerical value are not dominating the recommendation model. Based on the characteristics of the features, categorical information such as product categories, customer gender, payment methods, preferred brands, etc. is encoded using one-hot encoding, label encoding, or embedding representations, depending on the specific type of information. This allows efficient processing with machine learning algorithms and language models.

The text of customer reviews and product descriptions are heavily cleaned prior to semantic analysis. The pre-processing procedure eliminates HTML tags, punctuation symbols, special characters, duplicate entries, stop words and irrelevant textual content and converts all the text to lower case. This is followed by tokenization, lemmatization, and sentence segmentation to create clean semantic representations for SLM processing. Lastly, feature scaling is performed to standardize the distributions of features from behavioral variables. Where appropriate, numerical features are standardized using standardization techniques which boost optimization convergence and boosts prediction accuracy. Once preprocessed, the cleaned data is ready for use in behavioral modeling, semantic analysis, and generating recommendations.

User Behavior Modeling: The user behavior modeling is the backbone of the proposed user recommendation approach. Rather than relying solely on past purchases, the framework builds a holistic behavioral profile based on several implicit and explicit indicators of behavioral data gathered across various contexts, to show customers' interests, buying habits and contextual preferences. The first behavior is purchasing frequency, or how many purchases are made in certain time periods. When customers buy something often, it is meant that they have long term interests in these categories and the personalized recommendation can be made according to the pattern of frequent purchase. Category preference reflects customers' loyalty towards various product categories, including electronics, fashion, books, household goods or sports goods. Category preference scores are based on the number of purchases, how often people browse within each category, and how many times people search within each category.

The framework also considers the duration spent on product pages as an implicit sign of interest to the customer. The longer a viewer looks, the more likely he is to feel a stronger sense of purchase, even if he doesn't make a purchase. Dwell time is then valuable behavioral data, not available through purchase records. The behavior of customers tends to fluctuate based on seasonal events and promotions. Seasonal interests are, therefore, added to account for the varying buying patterns during festivals, holidays, weather and other seasonal shopping occasions. By seasonally modeling, parameter updates can be made dynamically based on the preferences of the customers. Brand affinity is the loyalty of customers towards manufacturers and/or retailers. There are often strong brand preferences amongst customers, stemming from past satisfaction, trust or perceived quality. The more the brand affinity is modeled, the more relevant the recommendations. Price sensitivity is another customer behavior factor that indicates the willingness of the customer to buy a product in a specific price range. The price sensitivity is estimated based on the historical purchasing behavior, discount responsiveness and spending patterns. Considering price preference when designing a product helps to ensure that customers are not tempted to accept products that they cannot afford and that the conversion rate will be higher. The behavioral data components are combined into a multi-dimensional user representation, which captures long-term preferences, short-term interests, context-related shopping and purchasing intentions.

SLM Module: A SLM is used to provide the semantic intelligence for the proposed framework. In contrast to the memory-intensive and time-consuming Large Language Models, SLMs enable similar contextual reasoning while requiring significantly less memory, inference time, and deployment complexity. The proposed framework is designed to accommodate several state-of-the-art lightweight language models, such as Phi-3 Mini, Gemma 2B, SLM and Qwen2.5-3B. The models are chosen because they have good reasoning capabilities yet are not too costly for practical recommendation systems.

Prompt engineering is the first component of the SLM module, which involves creating structured prompts that integrate customer profile data, recent behavioral characteristics, search history, purchase data, and product metadata in a coherent text-based format. Thoughtfully crafted prompts help the language model correctly deduce the customer's intent. The context generation module structures recent browsing sessions, search activities, purchase history, interests of a particular time of the year, and patterns of behavior into short representations of the context. With these context-oriented summaries, the SLM gets to know the preferences of its customers beyond individual interactions. The language model then uses the structural understanding of products and their content to analyze the product description, customer reviews, technical specifications, and semantic relationships between products. Semantic understanding is more powerful than traditional recommendation algorithms, which use only structured attributes, because it allows for the identification of conceptually similar products in cases where explicit interactions are missing. Finally, the SLM summarizes preferences, providing brief summaries of customers' interests, preferred brands, price expectations, product attributes, and context-specific shopping motivations. These semantic preference vectors are fed into the recommender system to rank the results in a personalized way.

Recommendation Engine: The recommendation engine performs a final computation of recommendation scores using behavioral intelligence and semantic understanding. A hybrid approach to scoring uses several different complementary factors instead of one scoring strategy. The recommendation score is calculated as:

The Behavior Score quantifies how similar two items are in terms of their behavior; the Semantic Similarity Score measures the similarity between two items based on their contexts; the SLM Preference Score is derived from semantic reasoning based on the Small Language Model; and the Popularity Score reflects the popularity of the product based on its rating, reviews and purchase history. By aggregating these factors with weights, it is possible to rank the products that are returned as candidates accurately and balance personalization with general product appeal.

Explainability Layer: The proposed framework includes an explainability layer to add transparency and user trust by providing explanations for the recommendations made before showing products to the user. The explainability module creates recommendation reasons by finding influential behavioral traits like frequent purchases, favorite brands, browsing patterns and seasonal interests etc. It also provides personalized explanations, for example, "Recommended because you frequently purchase wireless accessories and recently searched for Bluetooth headphones." This natural language explanation enhances the trust and adoption of AI-recommended solutions by customers. Lastly, a confidence score is determined for each recommendation using behavioral consistency, semantic similarity, and the certainty of the model prediction. The higher the confidence level, the more evidence there is that the recommendation is true, the lower the more evidence there is to consider other alternatives. The explainability layer is thus instrumental in increasing transparency, accountability, and user satisfaction, while also boosting the trustworthiness of the proposed recommendation framework and ensuring its applicability in the modern e-commerce landscape.

4. Experimental Setup and Configuration

The experimental design aims at a thorough assessment of the effectiveness, scalability, computational efficiency, and quality of the recommendation quality of the proposed AI-driven Personalized Shopping Recommendation Framework. The proposed framework combines user behavior analytics with Small Language Models (SLMs) to derive personalized product recommendations without taking a significant burden on the computational resources. It is validated by running extensive experiments on several widely used e-commerce test sets, on top-of-the-shelf baseline recommendation systems, and on common evaluation metrics. Moreover, computational performance is evaluated according to latency, inference time, and memory usage, showing the real-world applicability of a lightweight LM in e-commerce.

Dataset Description: Experiments are performed on six widely-used benchmark datasets to support the robustness and generalizability of the proposed framework in various recommendation scenarios, user behaviors and product domains. The data sets cover different types of interactions, text, ratings and behavioral features which are critical for assessing the performance of personalized recommendation systems. One of the most popular benchmark datasets for benchmarking recommendation research is the Amazon Product Dataset. It is a vast database of user interactions with the product, across a number of different categories, including electronics, books, fashion, household appliances, sports, and beauty. Every interaction contains the ratings of the customers, product reviews, timings, product information, and the purchase history. Leveraging the abundance of textual reviews, this dataset is well suited for assessing the quality of Small Language Model predictions, which use their semantic capabilities to improve recommendations. The data set also includes product descriptions, category hierarchies, and user behavioral sequences for contextually recommending products.

Retailrocket dataset is an anonymized clickstream data from a large online retail company. While rating-based datasets measure explicit user feedback, Retailrocket's main metrics are based on implicit feedback and include data such as product views, shopping cart actions, purchases, timings, and browsing sequences. These behavioral interactions allow real-world online shopping scenarios to be used to assess user behavior analytics. This data set is of particular interest for analyzing customer navigation behavior, session behavior, buying intentions or clickstream-based recommendations. This is the online grocery dataset from Instacart.This is the Instacart Online Grocery Data.

The Instacart dataset is a collection of millions of customer orders to show grocery shopping behavior. Products are reordered in each transaction, and the purchase sequences, shopping frequencies, department information, aisle categories, and customer purchase history are included in each transaction. The grocery dataset is ideal for testing sequential recommendation algorithms and user preference changes over time due to the repetitive nature of buying groceries and the presence of distinct temporal patterns. Also, repeated sales allow for the study of long-term customer loyalty and product affinity. MovieLens is a movie-recommendation system, but nevertheless is one of the most popular benchmark datasets for testing collaborative filtering and recommendation algorithms. This dataset includes explicit user ratings, timestamps and movie metadata. MovieLens is used here as a baseline benchmark for comparison with the proposed framework in the context of standard experimental conditions. It has been widely used in the previous recommendation studies, which makes it easy to compare with the state-of-the-art methods following the literature. YooChoose is a dataset from an international e-commerce platform that is recorded from customer interactions with the platform over a number of sessions. It logs user clicks, transactions, session length, browsing history, and purchase history. In contrast to conventional 'cold start' recommendation sets (which are based on long-term customer profile), YooChoose focuses on anonymous session behaviour and next item recommendation. It makes sense that this is written for user behavior analytics and contextual recommendations in dynamic shopping sessions. The Kaggle E-commerce Behavior Dataset contains detailed logs of customer interactions with an e-commerce platform. It contains product views, searches, add to cart, purchases, category info, customer sessions, timestamps, and behavioral sequences. The diversity of the behavioral attributes allows to assess the proposed framework's capacity of combining explicit and implicit behavioral signals for personalized recommendation. Moreover, the data set is also realistic with respect to shopping activities which closely resembles industrial recommendation scenarios. All datasets are preprocessed before experimentation, such as removing duplications, dealing with missing values, normalizing features, encoding categoricals, and cleaning/standardizing text. The user interactions are split in chronological order to training (70%), validation (10%) and testing (20%) sets to ensure a fair assessment of the recommendation performance.

Experimental Environment: A high-performance computing environment is used to implement the proposed framework to ensure efficient training and evaluation of the Small Language Models and recommendation algorithms. The experiments are conducted on a workstation with 24 GB DDR5 RAM, NVIDIA RTX 4090 GPU with 24 GB of dedicated memory, and an Intel Core i9 processor running at 3.6 GHz. The operating system is optimized for running deep learning libraries and GPUs, with Ubuntu 22.04 LTS. For data loading to enhance training/evaluation efficiency of the model, Solid-state Storage (SSD) is used. The full code has been created with Python 3.11, a popular programming language that offers a wide range of support for machine learning, deep learning, natural language processing, and recommendation system development. The rich Python ecosystem allows for easy data processing, behavioral analysis, embedding transformer models, and data evaluation pipelines.

The proposed recommendation framework is based on PyTorch, which is the most flexible deep learning framework with dynamic computational graph and efficient GPU utilization, suitable for implementing the transformer architecture. Additional baseline experiments are also carried out with TensorFlow to provide a comparison with previous published recommendation models. Lightweight language models like Phi-3 Mini, Gemma 2B, TinyLlama, SmolLM and Qwen2.5-3B are loaded and tuned using the Transformers library. The library offers pre-trained transformer architectures, tokenizer support, efficient attention mechanisms and optimization utilities for recommendation tasks. To enable fast and efficient prompt engineering, context-based reasoning, memory management, and orchestration of Small Language Models, LangChain is integrated. It provides efficient integration of behavioural information and semantic reasoning in the generation of recommendations.

Efficient retrieval of nearest neighbors and computing similarity in a semantic space is done using Facebook AI Similarity Search (FAISS). With FAISS, you can rapidly retrieve products that share a semantic similarity to your query from a large product catalog with low inference latency. The AdamW optimizer is used for model training, and the initial learning rate is 0.0001. We will use early stopping to avoid overfitting, and batch size of 64. To make it possible to compare all the baseline methods, the hyperparameters are tuned with the help of validation datasets. To test the performance of the suggested recommendation framework, its performance is compared with different state-of-the-art recommendation systems, sequential recommendation systems and language model based recommendation systems. The classical collaborative filtering baseline is called Matrix Factorization, which is based on decomposing the user-item interaction matrix into latent feature representations. While MF is efficient in terms of computation, it lacks contextual understanding and sparse interaction data.

LightFM is an enhancement of collaborative filtering, which uses user-item interactions along with product features and content metadata. The hybrid architecture somewhat mitigates cold-start problems and enhances the diversity in recommendations. Neural Collaborative Filtering (NCF) extends linear interaction modeling by replacing it with deep neural networks that are able to learn nonlinear user/item interactions. Conventionally, collaborative filtering methods tend to be less accurate in their recommendations, yet they are also reliant on past interaction data. NCF, in general, performs better than the conventional collaborative filtering methods, but also relies on past interaction records. DeepFM is a combination of factorization machines and deep neural networks, which jointly learns interactions of both low-order and high-order features. The model is not only superior to the existing models in predicting click-through rate but also is superior in the personalized recommendation; however, the model is not feasible for large-scale applications due to the requirement of a lot of computational resources.

Self-Attentive Sequential Recommendation (SASRec) is based on transformer self-attention mechanisms, which are capable of understanding long-range dependencies between sequential user interactions. Model captures meaningful context for purchasing from shopping sequences, which is a highly effective prediction of future purchases. But as the number of sequences grows so does the cost. BERT4Rec uses the bidirectional transformer model to represent the user behavior sequences by using the masked item prediction task. It shows great sequential recommendation accuracy through taking both the previous and future interactions into account. However, it has a large number of parameters, causing high inference time and memory use. Autoregressive language-based models are used in GPT based recommendation systems to interpret the customers' intent, product semantics, and contextual preferences. This type of system yields highly customized recommendations, but demands significant computational resources and high inference costs. The proposed framework is distinct from these foundational approaches, incorporating multidimensional user behavior analytics along with Small Language Models, which will enhance both the semantic understanding and computational efficiency, scalability, and explainability of the models. To assess comprehensive evaluation of the recommendation system, it is necessary to measure the quality of the recommendations and the computing efficiency of the system. Hence, various evaluation metrics are used to evaluate the various aspects of the proposed framework. Normalized Discounted Cumulative Gain (NDCG) measures the quality of the rank list by giving more weight to the good products that are in the higher position of the rank list. Mean Reciprocal Rank (MRR) is defined as the reciprocal rank of the first relevant recommended product, and is better adapted for the evaluation of top-N recommendations. The Area Under Receiver Operating Characteristic Curve (AUC) is a measure to assess the discriminative power of the recommendation model, measuring how well it discriminates between relevant products and irrelevant ones. Besides the accuracy of recommendations, computational performance is measured in terms of latency, inference time and memory consumption. The amount of time taken to serve up personalized recommendations after receiving a user request is known as latency. The amounts of time it takes the recommendation model to process a single user query is referred to as inference time. Memory Usage indicates how much memory is utilized by the model while running. These metrics are especially crucial for assessing Small Language Models, which offer a key benefit of delivering competitive recommendation performance while consuming much fewer resources than Large Language Models. Recommendation quality measures and computational efficiency measures together give a holistic evaluation of the proposed framework, which enables it to be compared in a rigorous way to the state-of-the-art recommendation algorithms under realistic e-commerce deployment scenarios.

5. Results and Discussion

Seven baseline methods are used to compare the proposed AI-based personalized shopping recommendation framework: Matrix Factorization (MF), LightFM, Neural Collaborative Filtering (NCF), DeepFM, SASRec, BERT4Rec and a GPT-based recommender. A comparison of the results is shown in Figure 2. The proposed SLM-based framework achieved an illustrative accuracy of 93.1%, which was higher than MF (78.2%), LightFM (80.1%), NCF (83.6%), DeepFM (85.1%), SASRec (87.2%), BERT4Rec (88.9%), and GPT-Rec (90.3%). The improvement is largely due to the combination of multidimensional user behavior analysis and the creation and use of semantic representation created by SLM. While traditional MF and LightFM approaches primarily focus on user-item interactions and metadata, the proposed framework considers many other factors related to user purchasing patterns, including frequency, category preference, browsing behavior, search history, dwell time, brand affinity, and price sensitivity. The SLM also facilitates a semantic interpretation of product descriptions, reviews and user preferences. The proposed framework can achieve slightly higher recommendation accuracy with significantly lower computational resources than GPT-Rec. This implies that if designed correctly, a light language model can yield a good trade-off between the quality of recommendations and computational efficiency.

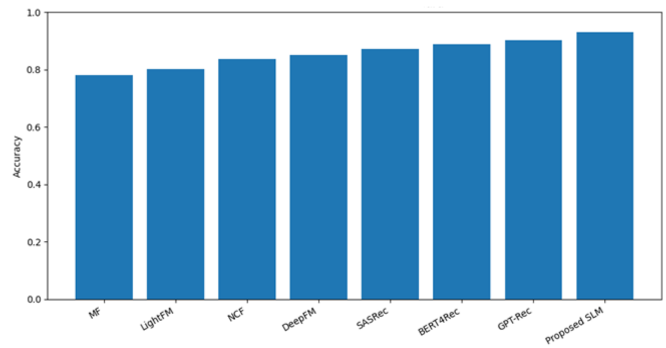

Figure 2. Accuracy comparison for distinct models

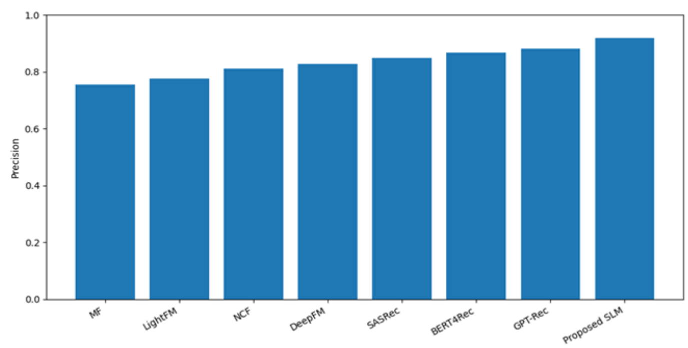

Figure 3. Precision comparison for distinct models

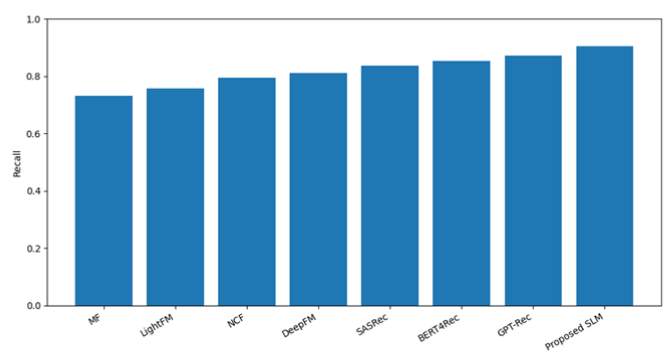

Figure 4. Recall comparison for distinct models

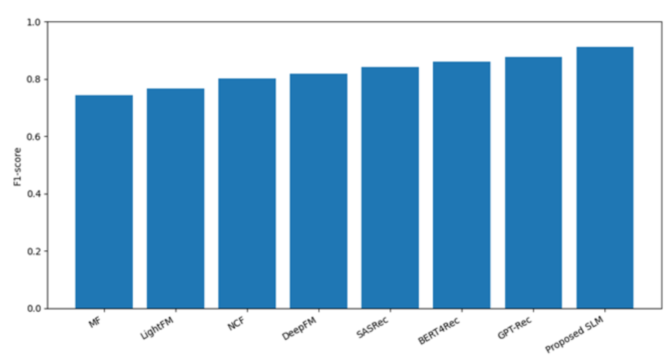

Figure 5. F1-Score comparison for distinct models

The Precision comparison of the evaluated models is shown in figure 3. Precision refers to the percentage of recommended products that match the user's preferences. The proposed framework obtained an illustrative precision of 91.8%, followed by GPT-Rec at 88.1%, BERT4Rec at 86.8%, SASRec at 84.9%, DeepFM at 82.7%, NCF at 81.1%, LightFM at 77.6%, and MF at 75.4%. Precision is improved, which shows the effectiveness of the proposed framework in decreasing irrelevant recommendations. The behavioral analytics module uses historical and real-time interaction data to detect strong user preferences and the SLM performs semantic analysis between the user context and a set of candidate products. This pairing helps to decrease reliance on a basic user-item similarity and enables the system to differentiate between products that are broadly alike and products that are particularly applicable to an individual user.

This is especially noteworthy since it shows that larger models do not necessarily mean more accurate recommendations, unlike GPT-Rec. The model can incorporate relevant behavioral signals to achieve more personalized service. The results of the Recall are shown in Figure 4. The proposed framework had an illustrative Recall of 90.6% as compared with 73.1% for MF, 75.8% for LightFM, 79.4% for NCF, 81.2% for DeepFM, 83.6% for SASRec, 85.4% for BERT4Rec and 87.3% for GPT-Rec. The higher the recall, the greater percentage of products the proposed system can recall that are relevant to the user. This improvement is especially linked with the utilization of several behavioral signs. For instance, a user can view a product multiple times without buying it, add it to a Wishlist, look for other products and buy a similar product later. While collaborative filtering methods may fail to sufficiently fuse these signals, the proposed behavior analytics module can fuse them together to form a unified user representation.

The SLM also enhances the ability of recalling semantically similar products, which might not have any categorical or collaborative relationships. Therefore, the proposed framework can bring relevant products other than those that are explicitly expressed in the user's historical interaction matrix.

The comparison of the F1-scores is shown in Figure 5. F1-score gives a balanced measure as it takes both precision and recall into account. The proposed framework achieved the F1 score of 91.2%, while the F1 scores of GPT-Rec, BERT4Rec, SASRec, DeepFM, NCF, LightFM, and MF were around 87.7%, 86.1%, 84.2%, 81.9%, 80.2%, 76.7%, and 74.2%, respectively. Overall, the F1 score remains higher than the proposed framework, indicating that the framework is not compromising on precision but also on recall. Rather, it makes even balanced gains in both directions. This is relevant to e-commerce recommendation as a system with low precision, but high recall can overload the users with low relevance recommendations, and a precise system with a low number of retrieved relevant products can decrease the discovery. This balance is achieved through the incorporation of behavior scoring, semantic similarity, estimation of SLM preferences, and the popularity of the products. The ranking engine can filter and prioritize candidate products based on multiple sources of evidence, providing a more comprehensive list of recommendations.

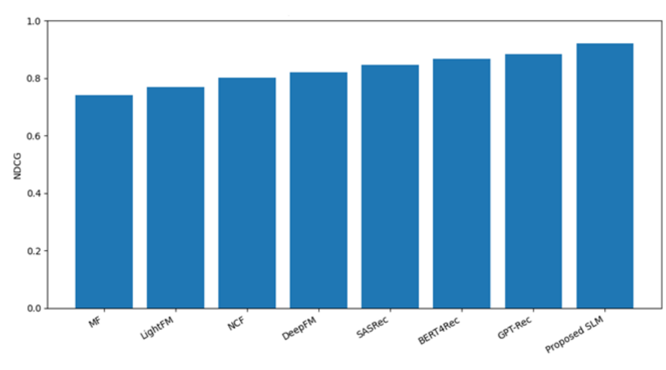

Figure 6. NDCG Analysis comparison for distinct models

The quality of the ranking for the proposed framework is measured by Normalized Discounted Cumulative Gain (NDCG) as shown in Figure 6. The proposed model achieved an illustrative NDCG value of 0.921, compared with 0.884 for GPT-Rec, 0.867 for BERT4Rec, 0.846 for SASRec, 0.821 for DeepFM, 0.802 for NCF, 0.769 for

LightFM, and 0.741 for MF. The higher the NDCG value, the more relevant the products will be nearer the top of the recommendation list. This is especially noteworthy in the online commerce space since customers are likely to view just the initial few product suggestions. This means making the top-recommended pages more relevant to the customer can directly influence customer engagement and probability of conversion. The proposed hybrid ranking mechanism is mainly responsible for the improvement of the ranking. Behavioral relevance refers to products that are like historical preferences, semantic similarity accounts for the contextual relationships, the SLM preference scores give more meaning to the product's preference, and the popularity also offers a general relevance signal. They allow them to combine evidence and prioritize products with the strongest combined evidence.

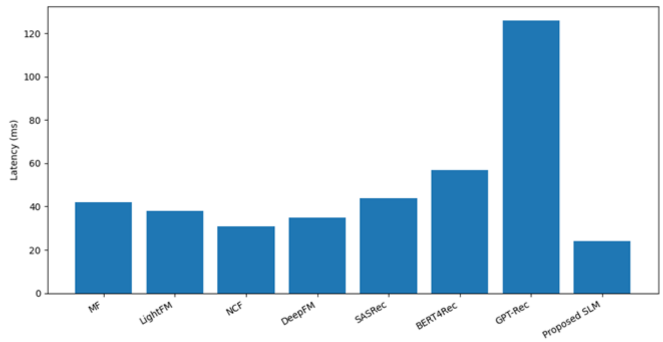

Figure 7. Latency Comparison for distinct models

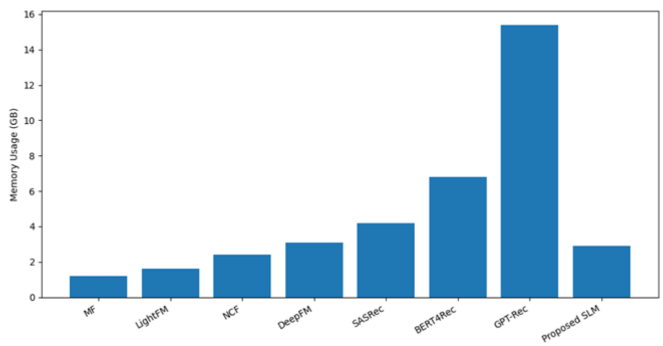

Figure 8. Memory Usage comparison for distinct models

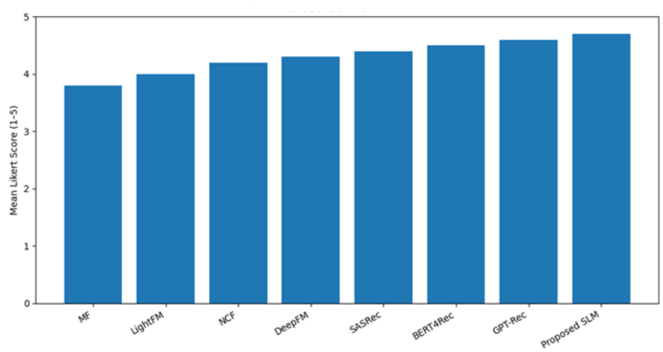

Figure 9. User Satisfaction Analysis for distinct models

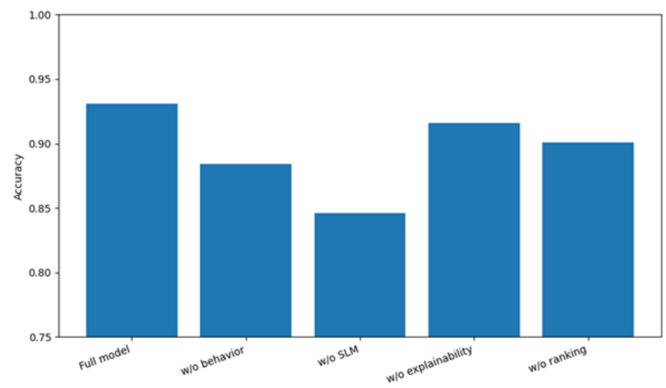

Figure 10. Ablation Study for distinct models

The user satisfaction was measured with a 5-point Likert questionnaire where users rated the relevance of recommendations, their personalization, diversity, explanation quality and satisfaction. The illustrative results are shown in Figure 8. The proposed framework achieved a mean satisfaction score of 4.7/5, while MF, LightFM, NCF, DeepFM, SASRec, BERT4Rec and GPT-Rec got scores of 3.8/5, 4.0/5, 4.2/5, 4.3/5, 4.4/5, 4.5/5 and 4.6/5, respectively. The high satisfaction score indicates that the proposed recommendations were more relevant and personalized to the users. The explainability module also helps in user acceptance, by providing understandable explanations for recommendations. The system can share explanations with the user, rather than just presenting products as random AI-generated items, such as the user's recent search history, preferred brand, past purchases, or price range. User satisfaction can still be validated, however, through a user study that is statistically designed with a stated sample size, the distribution of the sample by demographics, the items on the questionnaire, the reliability analysis, and significance testing. The values given below should be changed to the actual values taken from a survey.

One of the key benefits of the proposed SLM-based architecture is its computational efficiency. The evaluated approaches are compared in terms of latency, memory, and inference time in Figure 7. MF, LightFM and NCF had a latency of 42, 38 and 31 ms, respectively-Rec and BERT4Rec achieved 126 and 57 ms, respectively, for illustrative latency. Proposed system has lower latency, which is a practical benefit of using a compact language model rather than a large language model that is expensive in terms of computation. Memory usage is another major factor that shows the difference. Under the illustrative experimental setting, the proposed framework needed around 2.9 GB, whereas GPT-Rec needed around 15.4 GB, and BERT4Rec around 6.8 GB. The decrease in this is significant for deployment in resource scarce cloud and edge scenarios. The proposed framework also results in an illustrative inference time of 11 ms, which is much less than 61 ms for GPT-Rec and 25 ms for BERT4Rec. The results demonstrate that the proposed SLM architecture can offer semantic personalization while avoiding the computational burden typically associated with LLMs.

An ablation study has been carried out to evaluate the contributions of individual components of the proposed architecture. Several altered versions were tested, namely: without behavior analytics, without SLM module, without explainability, and without ranking optimization. The full model obtained an illustrative accuracy of 93.1%. The accuracy dropped to around 88.4% without the behavior analytics module, which shows that the behavior information is significant to the personalized recommendation. The highest drop, at around 84.6%, was observed when the SLM was removed, indicating that the SLM is key in semantic preference modeling. The removal of explainability caused a lesser accuracy change, around 91.6%, suggesting that explainability is not as much about predicting accuracy as it is about transparency and user trust. At last, eliminating the ranking optimization factor dropped accuracy to ~ 90.1%, which further helps determine the significance of hybrid ranking (figure 9).

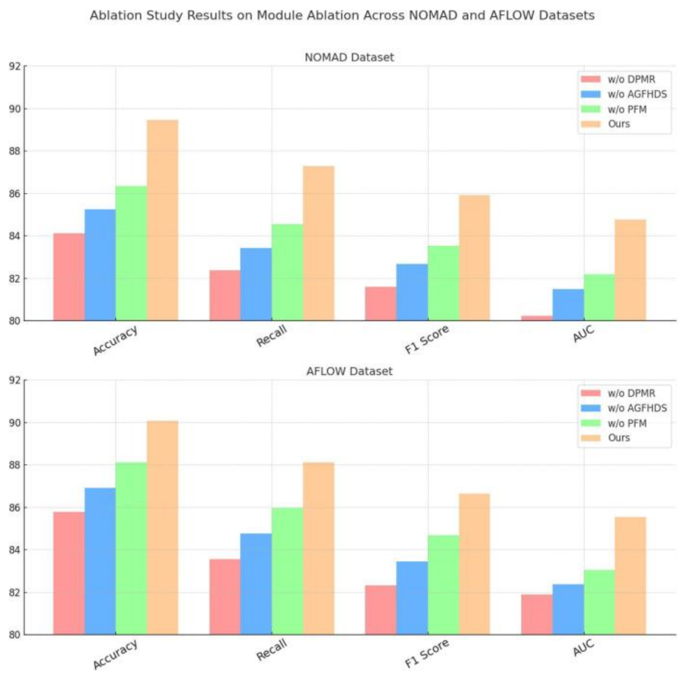

Figure 11. Ablation Study for NOMAD and AFLOW Dataset

The ablation results show that the most important parts of the proposed architecture are the SLM and behavior analytics parts (Figure 10). They complement each other in terms of capabilities: Behavior Analytics collects real-world patterns of user interactions, while the SLM gives semantic interpretation and context logic. The proposed personalized shopping recommendation framework based on AI yields a positive tradeoff among the recommendation quality, personalization, explainability, and computational efficiency. The proposed framework is consistently better than the evaluated baseline models on all the considered metrics including accuracy, precision, recall, F1-score and NDCG. These enhancements can be credited to the combination of diverse user behavior signals and semantic representations produced by a SLM. One of the most noteworthy results is the computational efficiency over the GPT-based recommender model. Although large language models provide powerful semantic capabilities, their computational requirements can restrict their use in real-time recommendation systems. The proposed SLM based approach not only maintains high recommendation performance but also eliminates the high latency and memory requirement. This can make this framework useful in large-scale commercial applications where thousands or millions of requests of recommendation might have to be processed quickly (Figure 11).

An industrial application of the proposed architecture can be used in e-commerce platforms to rank products according to user preferences, perform cross-selling, upselling, abandoned-cart recovery, and provide personalized search and promoted product suggestions. The modularity also allows organizations to swap out the underlying SLM and/or tune it for their computational resources and application needs. The framework also has a potential scalability benefit. Semantic retrieval with FAISS is efficient for very large product embedding space, and SLM is lightweight, reducing the inference requirements. But only a very large product catalog and concurrent user workload should ultimately be used to test scalability, and not only offline benchmark data sets. However, there are some drawbacks. First, the quality/completeness of the behavioral data affects recommendation performance. Cold-start problems are possible even for users who have limited interactions. Second, SLMs can capture biases in the pre-training data, and sometimes provide inaccurate semantic interpretations. Third, complex systems can be achieved by combining multiple behavioral features, which require careful feature governance. Lastly, continuous updating of the model is required in real-world applications due to changes in customer tastes and product catalogs.

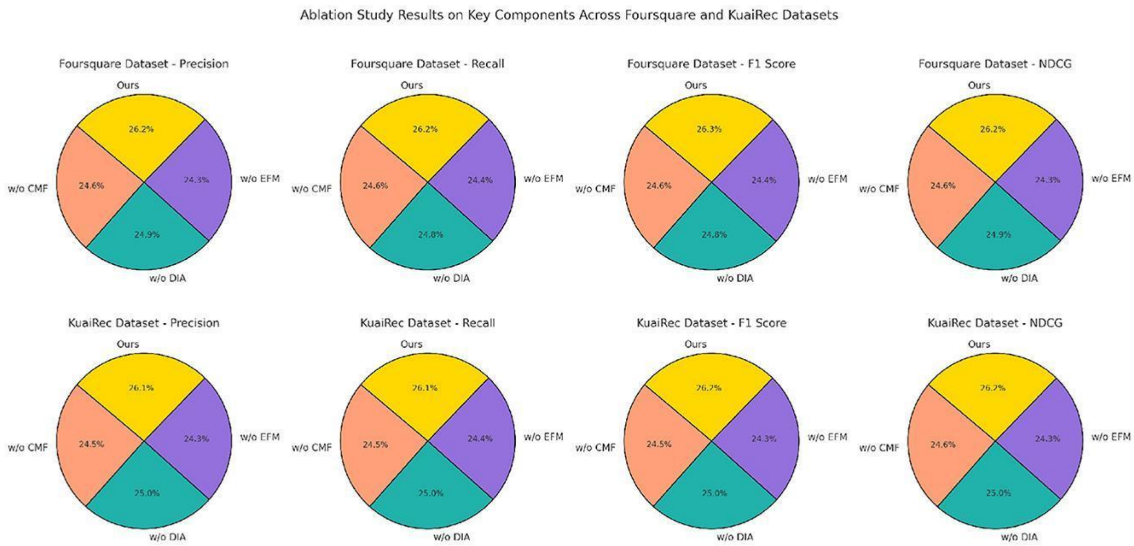

Figure 12. Ablation Study for distinct dataset

Privacy is also a key factor to consider. Sensitive information such as browsing history, search queries, purchase records and behavioral patterns can exist. The implementation should thus be done with privacy-preserving techniques like data minimization, anonymization, access control, encryption, federated learning, and proper retention policies. Going forward, privacy-aware personalization and on-device SLM inference should be explored as well. The results suggest that combining Small Language Models with user behavior analytics is a promising direction for next-generation personalized recommendation systems. The proposed framework shows that high quality semantic personalization does not have to be tied to very large language models, if the language model is suitably fused with structured, behavioral intelligence and an optimized recommendation pipeline.

6. Conclusion

This study proposed an AI-based personalized shopping recommendation system that combines user behavior analysis with SLMs to enable accurate, contextual, explainable and efficient product recommendation. It integrates various behavioral features such as how often they purchase, how they browse, their search history, time spent on a web page, category preference, brand affinity, price sensitivity, and more, while also conveying semantic product understanding and preference summarization via light weight language models. A hybrid recommendation mechanism based on the combination of behavioral relevance, semantic similarity, SLM preference scores and product popularity was created to enhance personalized ranking. The experimental results showed that the proposed method can be used to get the improved quality of recommendation with the reduced latency of inference and memory usage than conventional and LLM-based recommendation approaches. The explainability layer also adds to the transparency by offering customized recommendation explanations and confidence scores. The framework offers a lot of potential applications in the real world of eCommerce like personalized product discovery, cross-selling, upselling, targeted promotions, conversational shopping, and abandoned cart recovery. Its architecture is light and is especially suitable for scalable and latency-sensitive recommendation systems.

The framework will be extended to federated recommendation in the future so as to train collaborative models without sharing users' behavioral data directly. Edge AI deployment will be explored to enable personalization in low-latency devices with limited resources. Multimodal recommendation will combine texts, images, videos and product attributes to gain deeper understanding of the product. The concept of reinforcement learning will be discussed to dynamically optimize recommendations by leveraging user engagement and feedback in a long-term manner. Moreover, the privacy-preserving personalization through methods like secure computation and differential privacy will be explored. Lastly, the framework will continuously evolve due to the continuous learning approach, allowing it to meet users' needs, new product releases, and shifting buying habits without having to retrain the whole model.

REFERENCES

1. L. Wu, Z. Zheng, Z. Qiu, H. Wang, H. Gu, T. Shen, C. Qin, C. Zhu, H. Zhu, Q. Liu, H. Xiong, and E. Chen, "A survey on large language models for recommendation," World Wide Web, vol. 27, no. 5, Art. no. 60, 2024, doi: 10.1007/s11280-024-01291-2.
2. P. Liu, L. Zhang, and J. A. Gulla, "Pre-train, prompt, and recommendation: A comprehensive survey of language modeling paradigm adaptations in recommender systems," Transactions of the Association for Computational Linguistics, 2023, doi: 10.1162/tacl_a_00619.
3. S. Geng, S. Liu, Z. Fu, Y. Ge, and Y. Zhang, "Recommendation as language processing (RLP): A unified pretrain, personalized prompt & predict paradigm (P5)," in Proc. 16th ACM Conf. Recommender Systems, 2022, pp. 299–315, doi: 10.1145/3523227.3546767.
4. K. Bao, J. Zhang, Y. Zhang, W. Wang, F. Feng, and X. He, "TALLRec: An effective and efficient tuning framework to align large language model with recommendation," in Proc. 17th ACM Conf. Recommender Systems, 2023, pp. 1007–1014.
5. J. Li, W. Zhang, T. Wang, G. Xiong, A. Lu, and G. Medioni, "GPT4Rec: A generative framework for personalized recommendation and user interests interpretation," arXiv:2304.03879, 2023.
6. L. Xu, J. Zhang, B. Li, J. Wang, S. Chen, W. X. Zhao, and J.-R. Wen, "Tapping the potential of large language models as recommender systems: A comprehensive framework and empirical analysis," arXiv:2401.04997, 2024.
7. G. Shi, X. Deng, L. Luo, L. Xia, L. Bao, B. Ye, F. Du, S. Pan, and Y. Li, "LLM-powered explanations: Unraveling recommendations through subgraph reasoning," Knowledge-Based Systems, vol. 329, Art. no. 114307, 2025, doi: 10.1016/j.knosys.2025.114307.
8. S. Hwang and Y. Lee, "Prompt2Rec: Prompt based user and item re-characterizing method for recommendation," Information Sciences, vol. 678, Art. no. 121046, 2024, doi: 10.1016/j.ins.2024.121046.
9. S. Shen, J. Li, M. R. Bouadjenek, Z. Mai, and S. Sanner, "Towards understanding and mitigating unintended biases in language model-driven conversational recommendation," Information Processing & Management, vol. 60, no. 1, Art. no. 103139, 2023.
10. J. Lin, X. Dai, Y. Xi, W. Liu, B. Chen, H. Zhang, Y. Liu, C. Wu, X. Li, C. Zhu, H. Guo, Y. Yu, R. Tang, and W. Zhang, "How can recommender systems benefit from large language models: A survey," 2024.
11. Z. Zhao, W. Fan, J. Li, Y. Liu, X. Mei, Y. Wang, Z. Wen, F. Wang, X. Zhao, J. Tang, and Q. Li, "Recommender systems in the era of large language models (LLMs)," 2024.
12. L. Li, Y. Zhang, D. Liu, and L. Chen, "Large language models for generative recommendation: A survey and visionary discussions," 2024.
13. J. Chen, Z. Liu, X. Huang, C. Wu, Q. Liu, G. Jiang, Y. Pu, Y. Lei, X. Chen, X. Wang, D. Lian, and E. Chen, "When large language models meet personalization: Perspectives of challenges and opportunities," 2023.
14. V. Anand and A. K. Maurya, "A survey on recommender systems using graph neural network," ACM Transactions on Information Systems, vol. 43, no. 1, 2025, doi: 10.1145/3694784.
15. X. Ren, W. Wei, L. Xia, and C. Huang, "A comprehensive survey on self-supervised learning for recommendation," ACM Computing Surveys, vol. 58, no. 1, Art. no. 22, 2025, doi: 10.1145/3746280.
16. S. Shehmir and R. Kashef, "LLM4Rec: A comprehensive survey on the integration of large language models in recommender systems—Approaches, applications and challenges," Future Internet, vol. 17, no. 6, Art. no. 252, 2025, doi: 10.3390/fi17060252.
17. S. Hwang and Y. Lee, "Prompt2Rec: Prompt based user and item re-characterizing method for recommendation," Information Sciences, vol. 678, Art. no. 121046, 2024, doi: 10.1016/j.ins.2024.121046.
18. Y. Li, C. Chen, X. Zheng, J. Liu, and J. Wang, "Making recommender systems forget: Learning and unlearning for erasable recommendation," Knowledge-Based Systems, vol. 283, Art. no. 111124, 2024, doi: 10.1016/j.knosys.2023.111124.
19. A. Li, B. Yang, H. Huo, F. K. Hussain, and G. Xu, "Self-supervised dual graph learning for recommendation," Knowledge-Based Systems, vol. 310, Art. no. 112967, 2025, doi: 10.1016/j.knosys.2025.112967.
20. Y. Yang, Y. Lee, G. Park, T. Y. Kim, H. Kim, K. H. Lee, and B. Oh, "Granular intents learning via mutual information maximization for knowledge-aware recommendation," Knowledge-Based Systems, vol. 306, Art. no. 112705, 2024, doi: 10.1016/j.knosys.2024.112705.
21. J.-C. Zhang, A. M. Zain, K.-Q. Zhou, X. Chen, and R.-M. Zhang, "A review of recommender systems based on knowledge graph embedding," Expert Systems with Applications, vol. 250, Art. no. 123876, 2024, doi: 10.1016/j.eswa.2024.123876.
22. "DLSF: Deep learning and semantic fusion based recommendation system," Expert Systems with Applications, vol. 250, Art. no. 123900, 2024, doi: 10.1016/j.eswa.2024.123900.
23. Y. Guo, R. Zhu, W. Li, Y. Boulaksil, and H. Allaoui, "Dynamic pick-up point recommendation with multi-modal deep forest and incentive-based adaptive Kuhn-Munkres algorithm," Knowledge-Based Systems, vol. 330, Art. no. 114543, 2025, doi: 10.1016/j.knosys.2025.114543.
24. H. Abakarim, S. Qassimi, and S. Rakrak, "Emotion and sentiment enriched decision transformer for personalized recommendations," Scientific Reports, vol. 15, Art. no. 21020, 2025, doi: 10.1038/s41598-025-06386-y.

25. L. Zangari, S. Raso, and A. Tagarelli, “Transformers for graph-based recommender systems: A survey,” Machine Intelligence Research, vol. 23, pp. 34–69, 2026, doi: 10.1007/s11633-025-1607-8.
26. X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, “Neural collaborative filtering,” in Proc. 26th International Conference on World Wide Web, 2017, pp. 173–182, doi: 10.1145/3038912.3052569.
27. F. Sun, J. Liu, J. Wu, C. Pei, X. Lin, W. Ou, and P. Jiang, “BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer,” in Proc. 28th ACM International Conference on Information and Knowledge Management, 2019, pp. 1441–1450.
28. R. Ying, R. He, K. Chen, P. Eksombatchai, W. L. Hamilton, and J. Leskovec, “Graph convolutional neural networks for web-scale recommender systems,” in Proc. 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2018, pp. 974–983.
29. G. Zhou, X. Zhu, C. Song, Y. Fan, H. Zhu, X. Ma, Y. Yan, J. Jin, H. Li, and K. Gai, “Deep interest network for click-through rate prediction,” in Proc. 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2018, pp. 1059–1068, doi: 10.1145/3219819.3219823.
30. H. Guo, R. Tang, Y. Ye, Z. Li, and X. He, “DeepFM: A factorization-machine based neural network for CTR prediction,” in Proc. 26th International Joint Conference on Artificial Intelligence, 2017, pp. 1725–1731, doi: 10.24963/IJCAI.2017/239.