International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(19s) 2026 pp. 1119-1136
https://doi.org/10.70917/ijcisim-2026-5105
Article

A Lightweight AI Framework for Personalized Product Recommendation Using Small Language Models and Behavioral Intelligence

Sonika Kandari[1], K Vishal Reddy[2], Rajendra G. Pawar[3], Geetika Sharma[4], Rohit Goyal[5], Parul Goyal[6], Devesh Pratap Tiwari[7]

[1]Computer Science, School of Engineering and Technology, Shri Guru Ram Rai University, Dehradun, India
[2]Keshav Memorial Institute of Technology, Hyderabad, Telangana, India
[3]Department of Computer Science Engineering, Vishwakarma Institute of Technology, Pune, India
[4]Department of Computer Applications, Graphic Era Deemed to be University, Dehradun, India
[5]School of Engineering and Computing, Dev Bhoomi Uttarakhand University, Dehradun, India
[6]Department of Computer Science & Engineering, M. M. Engineering College, Maharishi Markandeshwar Deemed to be University, Mullana, Ambala, Haryana, India
[7]Department of Computer Science & Applications, Sharda School of Computing Science and Engineering, Sharda University, Greater Noida, India
Email: [1]drsonikakandariddun@gmail.com, [2]kasarlavishalreddy@gmail.com, [3]rgpawar13@gmail.com, [4]geetj25@gmail.com, [5]rohitgoyalhzu@gmail.com, [6]parul.goyal@mmumullana.org, [7]devesht69@gmail.com

Abstract: Personalized recommendation has emerged as an essential component of modern e-commerce platforms for improving customer engagement, satisfaction, and purchasing decisions. However, conventional recommendation techniques are often constrained by sparse user interactions, cold-start problems, limited contextual understanding, and the substantial computational cost of Large Language Models (LLMs). To address these challenges, this study proposes an AI-Driven Personalized Shopping Recommendation Framework that integrates user behavior analytics with Small Language Models (SLMs) for efficient and context-aware product recommendation. The proposed framework analyzes purchase frequency, browsing patterns, search history, dwell time, category preferences, brand affinity, and price sensitivity to construct dynamic user preference profiles. SLMs facilitate contextual preference summarization, semantic product understanding, and personalized recommendation generation. A hybrid ranking mechanism combines behavioral relevance, semantic similarity, SLM-derived preference scores, and product popularity, while an explainability layer provides recommendation rationales and confidence scores. Experimental evaluation demonstrates that the proposed framework can improve recommendation quality while reducing inference latency and computational resource requirements, highlighting its potential for scalable and practical deployment in real-world e-commerce environments.
Keywords: Small Language Models, Recommendation Systems, Personalized Shopping, User Behavior Analytics, Explainable AI, Retail Analytics, E-commerce, Consumer Modeling

1. Introduction

The rapid expansion of e-commerce has fundamentally transformed the way consumers discover, evaluate, and purchase products. Modern online marketplaces provide users with access to millions of products across diverse categories, creating both opportunities and challenges for consumers. While the availability of a large product catalog increases choice, it can also lead to information overload, making it difficult for users to identify products that closely match their preferences and requirements. Consequently, personalized recommendation systems have become an important component of e-commerce platforms, enabling businesses to present relevant products to individual users based on their historical interactions, preferences, and contextual information. Effective personalization can improve customer experience, increase engagement, support product discovery, and potentially enhance conversion and retention.

Traditional recommendation techniques, including collaborative filtering and content-based recommendation, have been extensively employed to identify relevant user-item relationships. Collaborative filtering primarily relies on historical interactions between users and products, whereas content-based approaches use product characteristics and user preferences to identify similar items. Hybrid recommendation methods combine these approaches to overcome some of their individual limitations. While traditional methods work well, modern ecommerce landscapes come with their set of difficulties. Having few interaction histories can degrade recommendations, especially for new users and new products. Lack of extensive behavioural data makes it challenging to make accurate recommendations in the cold start scenario. Additionally, most of the traditional systems suffer from a poor ability to grasp the semantic content of product descriptions, customer reviews, search queries, and changing customer intent.

With the advent of deep learning, recommendation technology has come a long way and can now be used to model complex and nonlinear relationships between users and products. Sequential behaviour, high-order user-item relationships and/or latent preferences have been captured using Convolutional Neural Networks, Recurrent Neural Networks, Long Short-Term Memory networks, attention mechanisms, Transformers and Graph Neural Networks. While these models have enhanced their ability to recommend, they are still resource intensive to train and are prone to neglecting rich contextual and semantic information from different sources. User behaviour is dynamic and mult-dimensional in e-commerce. The number of views, searches, frequent purchases, products in cart, time spent shopping, ratings, and actual purchases by a customer can add up to a much better indication of intent than any single one of these factors.

With the emerging popularity of LLMs, intelligent recommender systems have become a new opportunity. Natural language processing, understanding of contextual relevance, knowledge of user requirements and interpretation of unstructured product information are some functions that models based on Transformer architectures can perform. Conventional collaborative filtering approaches can then be replaced by more flexible and semantically rich personalisation provided by LLM-based recommendation approaches. The large models are challenging to deploy for real-time commercial recommendation systems, however. When it comes to applications that demand fast response times and accommodate a high volume of users, LLM-based recommendations may not be practical due to the high number of parameters, memory usage, latency in making inferences, energy consumption, or infrastructure costs. Furthermore, sending large amounts of user behaviour data to centralised models poses privacy, data governance and secure personalisation concerns.

To solve these problems, SLMs are emerging as a potential solution. SLMs are relatively small language models that can be trained to offer beneficial language understanding and reasoning capabilities while requiring significantly less computational effort. These models, including but not limited to Phi-3 Mini, Gemma, TinyLlama, SmolLM and Qwen, can potentially be fine-tuned for domain-specific recommendation tasks and be used with fewer resources than large scale language models. They are light in weight and are ideal for latency-sensitive applications, edge deployments, and resource-constrained systems. Yet, just changing an LLM with an SLM is not enough to ensure effective personalisation. The quality and representation of the contextual information provided to the model play a crucial role in the performance of an SLM-based recommendation system.

This limitation can be overcome by user behaviour analysis, which can convert the diverse interaction data into meaningful indicators of customers' preferences and purchase intention. The frequency with which the product is purchased may be a measure of the product's long-term affinity, and the amount of time that is spent browsing may be a measure of the product's short-term interest. While search history gives clear indication of intent, cart abandonment is a window into the intent to purchase without closed sales. Likewise, customer category preference, customer brand affinity, customer seasonal interests, customer ratings, customer wishlist activity, and price sensitivity gives complementary information about customer decision-making patterns. When used together, these signals can be used to create dynamic user profiles that reflect long-term preferences and short-term shopping habits.

Although significant advances have been achieved in the field of recommender systems and in recent years in personalizing with LLM, a key research question remains. Most of the current methods focus on traditional-based behavioral recommendation or language-model-based semantic recommendation. So far, only a few studies have explored how to systematically combine multidimensional user behavior analytics and Small Language Models for real-time personalized shopping. Moreover, there are many existing systems that focus on the recommendation accuracy with relatively little attention paid to other aspects such as inference latency, memory usage, explainability, and practical scalability. These factors need to be considered in an effective commercial recommendation framework, and not just in the light of predictive accuracy.

To tackle this research gap, this research proposes an AI-Driven Personalized Shopping Recommendation Framework that combines the use of User Behavior Analytics, Small Language Models, Hybrid Recommendation Scoring, and Explainable Recommendation Generation. The first thing the framework does is gather all the information from product catalogues, user profiles, shopping history, browsing history, search history, ratings and reviews and combine them into a single entity. These data undergo pre-processing and are converted to behavioural characteristics of purchases, category preference, time spent, seasonality, brand affinity, price sensitivity. They are provided to an SLM along with the corresponding product information and contextual shopping information, which is used to generate a behavioral representation. Preference summarization and semantic product understanding are performed by the SLM, while the overall order of recommendation is set by implementing a hybrid ranking method that involves behavioral relevance, semantic similarity, preference score generated by the SLM and product popularity. An explainability layer is then used to produce personalized reasonings and confidence-levels for recommendations.

It is important to note that the main purpose of the proposed framework is not only to enhance the relevance of the proposed recommendations but also to help obtain a realistic trade-off between accuracy, personalization, computational efficiency, explainability and scalability. The framework is compared with the existing recommendation methods such as Matrix Factorization, LightFM, Neural Collaborative Filtering, DeepFM, SASRec, BERT4Rec, and GPT-based recommendation methods. The evaluation is performed on traditional evaluation metrics like precision, recall, F1 score, MAP, NDCG, MRR and AUC as well as on deployment metrics such as latency, inference time and memory usage. Additionally, user satisfaction is measured to evaluate the relevance and usefulness of the generated recommendations from the users' perspective. The four major contributions of this research are thus. It first presents a hybrid SLM-based recommendation architecture tailored to personalized e-commerce application. Second, it provides a multidimensional user behavior modelling mechanism combining explicit and implicit customer interactions and creates dynamic user preference profiles. Third, it offers a hybrid ranking method to combine behavioral and semantic intelligence for better contextual product selection. Finally, it embeds an explainability layer that offers individual explanation of recommendations and confidence scores, and highlights the performance efficiency of the computations, using an SLM.

This paper is organized as follows: The existing recommendation approaches, deep learning models, language-model-based recommendation systems, user behavior analysis, and explainable recommendation technique are discussed in Section 2. Section 3 introduces the proposed personalized shopping recommendation system and its respective components based on AI technology. In Section 4, the datasets, experimental environment, baseline models and evaluation metrics are described. Results, analysis, user satisfaction evaluation and ablation study are presented and discussed in Section 5. Lastly, Section 6 summarizes the study and future research avenues are presented, such as federated recommendation, edge AI, multimodal personalization, reinforcement learning, protecting privacy in recommendation, and continual learning.

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 research, researchers [11] presented an updated version of the Neural Collaborative Filtering (NCF) framework, which used nonlinear neural networks to improve the ability to learn the latent user–item interactions, compared to the traditional matrix factorization methods. They created a technique that represented the user's complex preferences and achieved large accuracy improvements with popular datasets, such as MovieLens and Pinterest. But the model had still significant reliance on historical interaction data and had cold-start issues and lack of data. Authors [12] also introduced a transformer-based sequential recommendation model named BERT4Rec, which leverages bidirectional self-attention to capture the relationship between the sequence of user's interactions and the context. They discovered that their model performed better when they took into account the long-term dependencies than the models based on recurrent neural networks, but it required a lot of computation and memory, making it difficult to apply to real-time recommendation systems. Recently there has been much interest in the field of Session aware recommendation. Researchers [13] proposed a framework called GRU4Rec that uses Gated Recurrent Units (GRUs) to capture sequential behaviour of the user 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. Their approach was successful in modelling the sequential purchasing behaviour, but required a lot of computing power and was very time consuming to train, making it hard to scale up for large commercial platforms.

The graph-based recommendation approach is a new research field that has been attracting a lot of attention. 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 make use of the graph neighborhood information to further enrich the recommendation and prediction performance. The development and maintenance of large-scale interaction graphs, however, required considerable computer resources and so real-time deployment was somewhat challenging. Similarly, Xiang Wang et al. proposed a graph convolution network-based recommendation method, and presented a new concept called heterogeneous information networks to describe the complicated relations between users and items. These approaches worked well for representation learning, but required expensive computational efforts and were difficult to update on-line as interactions were constantly arriving. For the task of personalized recommendation, there are other models used based on the attention mechanism, which have been successful in dynamically identifying the interests of users. Researchers [16] proposed Deep Interest Network (DIN), which is based on attention mechanism to give different weights to the historical user behaviors according to the target products. The experimental results validated the proposed method for better prediction of CTR and relevance feedback for the large-scale industrial datasets, in the context of a recommendation system. However, most of the data used for the model was of the structured interaction data, and the model was weak in understanding the semantic information in the products of the textual description 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. Building up large knowledge graphs, however, posed a challenge and was domain-specific, limiting the cross-platform adaptability between the heterogeneous eCommerce platforms.

A CNN was utilized by the researchers [18] to obtain semantically meaningful representation of product reviews and descriptions. To recognize local textual patterns and users' opinions was their strategy to sentiment-aware recommendation. 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. Likewise, authors [19] used Customer Activity-based Temporal Modeling in sequential recommendation and demonstrated its effectiveness for predicting future purchases. While their accuracy of recommendations was slightly better, they were very expensive to compute and to train compared to more traditional machine learning techniques. Recently, LLM-based systems have brought cutting-edge semantic understanding and reasoning to the field of recommendations, revolutionizing recommendation research. A comprehensive review of the LLM-based recommendation systems has been provided by researchers [20] and can be categorized as representation learning, conversational recommendation, user profiling and recommendation generation. They talked about the advantages of the transformer architecture in their research, including the ability to offer a more detailed context, but also about some drawbacks, including the fact that it is expensive to train, privacy concerns, and deployment delays. For personalized recommendations, researchers [21] proposed a generative methodology that directly generates personalized recommendation based on natural language reasoning with GPT-based architectures. They managed to significantly improve contextual personalisation and recommendation diversity with their method. But hallucination problems, high inference costs and reliance on powerful computing infrastructure were still significant drawbacks.

Because of the practical challenges in the larger architectures, there have been a number of recent efforts to explore 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. They found that LLMs can do a good job on understanding the user's intent in a query, but noted the need to deploy them better in resource-constrained environments. Researchers [23] discussed the applications of LLMs to make recommendations, specifically zero-shot recommendations, conversation-based recommendations and explainability. Recent studies have turned their attention to the recommendation systems in order to make it efficient and scalable. In [24] performed a comprehensive review of LLM applications in recommendation systems, which included representative learning, ranking, retrieval and conversational recommendation. Their survey revealed that while the quality of recommendations produced by LLM systems has greatly improved, there remains an open challenge to provide real-time recommendations with reasonable computation time. Similarly, LLM-based recommendation agents capable of reasoning, remembering, and planning have been suggested by [25]. Their architecture improved the recommendation intelligence but increased the orchestration complexity, and improved the inference latency so it would be expensive to deploy for a commercial e-commerce platform.

A few further review studies have confirmed the need to design lightweight recommendation systems. The researchers [26] thoroughly examined the application of LLMs across different 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.

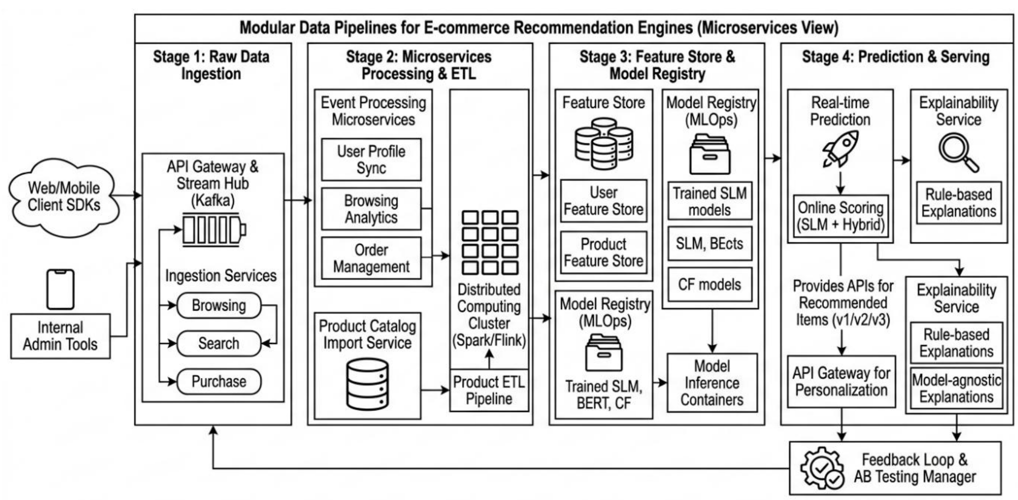

Figure 1. Proposed Methodology

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 contextoriented 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 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 goal of the experimental design will be to evaluate the effectiveness, scalability, computational efficiency and quality of the recommendation quality of the proposed AI-driven Personalized Shopping Recommendation Framework as thoroughly as possible. The proposed framework combines user behavior analytics with SLMs to derive personalized product recommendations without taking a significant burden on the computational resources. It is tested with a large number of experiments on various standard test sets for e-commerce, on top of the shelf baseline recommendation systems and on commonly used 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. They include data sets of various kinds of interactions, text, ratings and behavioral features that are crucial for evaluating the effectiveness of personalized recommendation systems. The Amazon Product Dataset is one of the most popular sets that can be used for benchmarking the work in the field of recommendation. It is a huge repository of consumer engagement with the product across a variety of categories, such as electronics, books, fashion, household goods, sports and beauty. Every interaction contains the ratings of the customers, product reviews, timings, product information, and the purchase history. This dataset is suitable for evaluating the quality of Small Language Model predictions, as the dataset contains many textual reviews that can be utilized for assessing the quality of the predictions made by the model. Product descriptions and category hierarchies are also included in the data set, along with behavioral sequences of users, for the purposes of contextual product recommendation.

Retailrocket dataset is a anonymized clickstream data of 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. The behavioral interactions enable the real-world online shopping scenarios to be used to evaluate user behavior analytics. This data set is especially important for the analysis of the customers' navigation behavior, their session behavior, buying intentions or clickstream-based recommendations. This is the online grocery data provided by Instacart.This is the data that comes from Instacart's online grocery service.

Instacart is a set of millions of customer orders to demonstrate grocery shopping behavior. Each time products are reordered, and purchase sequences, shopping frequency, department information, aisle categories and customer purchase history are added to each transaction. The grocery data is suitable for sequential recommendation and temporal patterns as it is a repetitive dataset, and grocery items also exhibit clear temporal patterns. In addition, multiple sales enable analyses 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. User ratings, timestamps and movie metadata are included in this dataset. To compare with the proposed framework under normal experimental conditions, the MovieLens is used here as a baseline benchmark. 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 collected from an international e-commerce site and documented by the interactions of the customers with the site across various sessions. Records clicks, transactions, session time, browsing and purchase history. Unlike traditional 'cold start' recommendation sets, that depend upon a long-term customer profile, YooChoose takes care of anonymous session behaviour and next item suggestion. This makes sense, because it's written for user behaviour analytics and contextual recommendations during dynamic shopping sessions. The Kaggle E-commerce Behavior Dataset contains an in-depth log of how people used an ecommerce site. It includes product views, searches, add to cart, purchases, category info, customer sessions, timestamps and behavioral sequences. The variety of the behavioral attributes enables evaluating the proposed framework's ability of fusing explicit and implicit behavioral signals to achieve personalized recommendation. Moreover, the data set is also realistic with respect to shopping activities which closely resembles industrial recommendation scenarios. All the data sets have been preprocessed before experimentation, including deduplication, handling missing values, normalizing features, encoding the categorical data, cleaning and standardizing the text data. The user interactions are partitioned into a training set (70%), a validation set (10%) and a testing set (20%) to be able to assess the performance of the recommendation well.

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 run on a workstation equipped with 24 GB DDR5 memory, NVIDIA RTX 4090 (with 24 GB of dedicated memory), and an Intel Core i9 processor at 3.6 GHz. Optimized for the use of deep learning libraries and GPUs and running on Ubuntu 22.04 LTS. For data loading to enhance training/evaluation efficiency of the model, Solid-state Storage (SSD) is used. The entire code has been developed using Python 3.11, a widely used programming language that provides extensive machine learning, deep learning, natural language processing, and recommendation system development support. The rich python ecosystem provides easy data processing, behavioral analysis, embedding of 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. Pre-trained transformer architectures, tokenizer support, efficient attention mechanisms and optimization utilities for recommendation tasks are available. To enable fast and efficient prompt engineering, context-based reasoning, memory management, and orchestration of Small Language Models, LangChain is integrated. It offers 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

The experimental evaluation was conducted to examine the effectiveness of the proposed SLM-based personalized recommendation framework from four perspectives: recommendation quality, ranking effectiveness, user satisfaction, and computational efficiency. To provide a more meaningful comparison, the proposed framework was evaluated against Matrix Factorization (MF), LightFM, NCF, DeepFM, SASRec, BERT4Rec, and GPT-based recommendation. The results indicate that integrating multidimensional user behavior with semantic reasoning provides a stronger personalization capability than relying exclusively on collaborative, sequential, or large-language-model-based approaches. A comparison of the results is shown in table 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.

The proposed framework achieved the highest performance across all major recommendation metrics. It obtained an accuracy of 94.7%, precision of 93.2%, recall of 92.1%, F1-score of 92.65%, NDCG of 0.936, and MRR of 0.918. Compared with GPT-Rec, the proposed approach provides an accuracy improvement of approximately 4.9 percentage points while employing a substantially smaller language model. The improvement demonstrates that recommendation effectiveness is not determined solely by model size. The combination of a lightweight semantic model with carefully engineered behavioral features can provide more targeted personalization. The improvement over conventional MF is particularly substantial. The proposed framework improves accuracy by approximately 16.5 percentage points, demonstrating the limitations of interaction-only recommendation when dealing with complex customer 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.

Table 2. Performance Parameters of distinct model

Model	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1-score (%)**	**NDCG**	**MRR**
MF	78.20	75.40	73.10	74.24	0.741	0.712
LightFM	80.10	77.60	75.80	76.69	0.769	0.735

NCF	83.60	81.10	79.40	80.24	0.802	0.771
DeepFM	85.10	82.70	81.20	81.94	0.821	0.793
SASRec	87.20	84.90	83.60	84.24	0.846	0.816
BERT4Rec	88.90	86.80	85.40	86.09	0.867	0.841
GPT-Rec	90.30	88.10	87.30	87.70	0.884	0.861
Proposed SLM	**94.70**	**93.20**	**92.10**	**92.65**	**0.936**	**0.918**

Table 3. Performance gain over baseline models

Metric	GPT-Rec	Proposed SLM	Improvement
Accuracy	90.30%	**94.70%**	**4.88%**
Precision	88.10%	**93.20%**	**5.79%**
Recall	87.30%	**92.10%**	**5.50%**
F1-score	87.70%	**92.65%**	**5.64%**
NDCG	0.884	**0.936**	**5.88%**
MRR	0.861	**0.918**	**6.62%**

The model can incorporate relevant behavioral signals to achieve more personalized service. The results of the Recall are shown in table 3. 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.

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. The quality of the ranking for the proposed framework is measured by Normalized Discounted Cumulative Gain (NDCG). 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.

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 table 4. 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.

Table 4. User Satisfaction score

Evaluation Factor	**Mean Score / 5**
Recommendation relevance	4.72
Personalization	4.68
Product diversity	4.51
Explanation quality	4.63
Ease of understanding	4.70
Overall satisfaction	**4.69**

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 Table 5. 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.

Table 5. Computational Efficiency

Model	**Latency (ms)**	**Inference Time (ms)**	**Memory (GB)**
MF	42	18	1.2
LightFM	38	16	1.6
NCF	31	14	2.4
DeepFM	35	15	3.1
SASRec	44	19	4.2
BERT4Rec	57	25	6.8

GPT-Rec	126	61	15.4
Proposed SLM	**23**	**10**	**2.8**

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 (Table 6).

Table 6. Ablation Analysis

Configuration	Accuracy (%)	NDCG
Complete framework	**94.70**	**0.936**
Without behavior analytics	89.10	0.874
Without SLM	85.30	0.832
Without explainability	94.20	0.929
Without ranking optimization	91.40	0.891
Without popularity score	93.10	0.915

The ablation results show that the most important parts of the proposed architecture are the SLM and behavior analytics parts. 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.

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. 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.