Design and Effectiveness Evaluation of an E-commerce Recommendation System Based on Large Language Models

Shuang Zhou*, Yun Liu

Beijing Information Technology College, Fengtai, Bejing 100070, China

*zhous@bitc.edu.cn

Abstract. Traditional e-commerce recommendation systems primarily rely on collaborative filtering, matrix factorization, or deep neural networks. This leads to significant shortcomings in scenarios such as cold starts, long-tail product recommendations, interest expansion, and alleviating information cocoons. In recent years, large language models (LLMs) have made remarkable progress in semantic understanding, knowledge reasoning, and generative expression, presenting new opportunities for e-commerce recommendation systems. This paper analyzes the typical architecture and limitations of traditional recommendation systems, as well as the current status and development trends of LLMs in the recommendation domain. It proposes a solution for an e-commerce recommendation system that integrates LLMs. This validates the feasibility and effectiveness of the proposed solution. Finally, the paper summarizes the limitations of the research and outlines future research directions.

Keywords: Large Language Model, User Interest Modeling, Explain ability, User Interest Modeling,

1 Introduction

With the rapid development of the internet and mobile internet, e-commerce has become an essential part of people's daily lives. According to statistical reports published by the China Internet Network Information Center (CNNIC), the number of online shopping users in China continues to grow, and the transaction volume of e-commerce maintains steady growth. On e-commerce platforms where massive products and users coexist, recommendation systems play a crucial role in helping users quickly discover products of interest and enhancing the shopping experience. They also provide strong support for platforms to increase user stickiness and boost sales [1].

Traditional e-commerce recommendation systems are mostly based on techniques such as collaborative filtering (CF), matrix factorization (MF), deep learning models (e.g., Wide & Deep, DeepFM, Neural Collaborative Filtering), and graph neural networks (GNN). These methods have alleviated the problem of information overload to some extent and achieved significant practical results. However, as user demands

J. Rak et al. (eds.), *Proceedings of the 2026 11th International Conference on Social Sciences and Economic Development (ICSSED 2026)*, Advances in Economics, Business and Management Research 391,
https://doi.org/10.2991/978-94-6239-701-9_65

become increasingly diverse and personalized, traditional recommendation systems have gradually exposed some limitations [2, 3].

The cold-start problem is also prominent. For new users or new products, due to the lack of sufficient historical interaction data, methods like collaborative filtering often struggle to generate accurate recommendations. Although some content-based recommendation methods can partially alleviate the cold-start issue, their effectiveness remains limited when facing new users due to the absence of user profiles[4].

The recommendation performance for long-tail products is also suboptimal. E-commerce platforms host a large number of long-tail products with low sales volumes. Although these products are diverse, their limited exposure opportunities make them difficult for users to discover. Explainability is also insufficient.

In recent years, large language models (LLMs) such as DEEPSEEK, GPT, BERT, and LLaMA have achieved breakthroughs in the field of natural language processing. By pre-training on large-scale corpora, LLMs have developed powerful semantic understanding, knowledge reasoning, and generative expression capabilities. Applying LLMs to e-commerce recommendation systems holds promise for addressing the aforementioned limitations [5, 6].

2 Research Status at Home and Abroad

2.1 Research Status of Traditional E-commerce Recommendation Systems

The development of traditional e-commerce recommendation systems has roughly gone through four stages. The first stage was collaborative filtering, one of the earliest techniques applied in recommendation systems, including user-based CF and item-based CF. The core idea of collaborative filtering is to recommend products liked by similar users or similar to those historically liked by the target user by calculating similarity between users or items.

The second stage was matrix factorization, which decomposes the user-item rating matrix into the product of user latent vectors and item latent vectors to capture potential features of users and items. Matrix factorization mitigates data sparsity to some extent and improves recommendation accuracy. Representative methods include SVD and SVD++. However, matrix factorization still struggles with cold-start issues and lacks clear semantic interpretations of user and item representations.

Recent developments have focused on deep learning-based recommendation systems. With advancements in deep learning, more studies have applied deep learning models to recommendation systems. Deep neural networks can automatically learn high-order feature representations of users and items, capturing complex nonlinear relationships. Representative deep recommendation models include Wide & Deep, DeepFM, Neural Collaborative Filtering (NCF), and YouTube DNN. These models have achieved better recommendation results than traditional methods in multiple public datasets and real-world applications.

The latest stage involves graph neural network-based recommendation systems. GNNs naturally model user-item interaction relationships as graph structures and use graph convolution operations to capture high-order neighbor information, thereby improving recommendation accuracy.

2.2 Research Status of LLM Applications in Recommendation Systems

In recent years, with the rapid development of LLMs, more studies have explored their application in recommendation systems. LLMs can be used for data augmentation by generating auxiliary information such as user profiles, product descriptions, and review summaries to expand training data and improve model generalization.

LLMs can also serve as text feature encoders, encoding text information such as user reviews and product descriptions to generate high-dimensional semantic vectors as input features for recommendation models. Some studies directly apply LLMs to rating prediction or ranking tasks, leveraging their semantic understanding and reasoning capabilities to generate recommendations. LLMs can generate natural language explanations for recommendations, improving explainability and user trust. For instance, the RecGPT framework introduces a specialized recommendation explanation LLM (LLM_RE) to attach personalized and understandable explanations to recommended products.

Despite progress, LLM applications in recommendation systems still face challenges such as high computational resource consumption, high inference latency, model hallucination, and data privacy protection. Therefore, designing efficient, reliable, and explainable e-commerce recommendation systems based on LLMs remains a critical area for further research.

3 Related Theoretical and Technical Foundations

3.1 Overview of Traditional E-commerce Recommendation Systems

The typical architecture of traditional e-commerce recommendation systems includes:

Data Collection Module: Collects user behavior data (e.g., browsing, clicking, favoriting, adding to cart, purchasing), product data (e.g., product ID, name, category, price, brand, description), user data (e.g., user ID, gender, age, region), and other auxiliary data (e.g., reviews, ratings, promotional activity information).

Data Preprocessing: Cleans, denoises, and normalizes collected data, addressing missing values and outliers, and converts raw data into formats suitable for model input. Features are extracted, including user features (e.g., user ID, gender, age, region, historical behavior statistics), product features (e.g., product ID, category, brand, price, sales, ratings), contextual features (e.g., time, location, device type), and cross features (e.g., user-product interaction features, user-category interaction features). Recommendation Algorithm Module: Generates recommendations based on extracted features using algorithms such as collaborative filtering, matrix factorization, deep learning, and GNNs.

3.2 Collaborative Filtering

Collaborative filtering is one of the earliest recommendation techniques, based on calculating similarity between users or items to recommend products liked by similar users or similar to those historically liked by the target user. It includes user-based CF and item-based CF.

User-based CF: Finds users with similar interests to the target user and recommends products liked by these similar users but not yet interacted with by the target user. Similarity metrics include cosine similarity and Pearson correlation.

Item-based CF: Finds products similar to those historically liked by the target user and recommends them. Similarity between items is calculated first, followed by recommending the most similar items.

While user-based CF is simple and intuitive, it suffers from data sparsity, cold starts, and scalability issues. Item-based CF mitigates data sparsity and is more scalable (due to fewer items than users) but still faces cold-start challenges for new products.

3.3 Matrix Factorization

Matrix factorization decomposes the user-item rating matrix into the product of user latent vectors and item latent vectors to capture potential features. Representative methods include SVD and SVD++.

SVD: Decomposes the user-item rating matrix Rinto R=UΣV, where Uis the user latent vector matrix, Σis the diagonal matrix, and Vis the item latent vector matrix. Low-dimensional latent representations are derived to predict ratings for unrated items.

SVD++: Improves SVD by incorporating implicit feedback (e.g., browsing, favoriting) to enhance accuracy.

Matrix factorization mitigates data sparsity and improves accuracy but struggles with cold starts and lacks semantic interpretability.

4 Design Scheme of an E-commerce Recommendation System Based on Large Language Models

4.1 Overall System Architecture Design

The overall architecture of the e-commerce recommendation system based on large language models comprises five core modules: data acquisition, data preprocessing, user interest modeling, recommendation generation, and recommendation justification.

The data acquisition module is responsible for gathering user behavioral data, product-related data, user demographic data, and supplementary auxiliary data. The data preprocessing module conducts cleaning, denoising, and normalization procedures on the collected data, and extracts user features, product features, contextual features, and cross-domain features.

In the user interest modeling module, a large language model is employed to semantically encode textual representations of user behaviors, thereby generating a user interest vector. The recommendation module synthesizes outputs from traditional collaborative filtering techniques, deep learning models, and the user interest vector to produce ranked recommendation lists. The recommendation justification module leverages the large language model to generate natural language explanations that articulate the rationale behind each recommendation, enhancing transparency for end users.

Specifically, the data acquisition module collects user behavioral records—including browsing, clicking, favoriting, adding to cart, and purchasing events—as well as product attributes (e.g., product ID, name, category, price, brand, and description), user profile information (e.g., user ID, gender, age, and location), and additional auxiliary data such as customer reviews, ratings, and promotional campaign details.

The data preprocessing module addresses data quality issues by handling missing values, removing noise, identifying and treating outliers, and normalizing data formats, thus transforming raw inputs into structures compatible with downstream modeling components. Feature extraction is performed to obtain user features, product features, contextual features, and cross features from the processed dataset.

4.2 User Interest Modeling Based on Large Language Models

The user interest modeling component applies a large language model to perform semantic encoding of user behavior text, resulting in a high-dimensional user interest vector. This approach enables the model to capture nuanced meanings within user queries, product descriptions, and review content, thereby uncovering latent user needs and preferences with greater precision. Consequently, the generated interest vectors contribute to more accurate and contextually relevant recommendation outcomes.

The recommendation module integrates traditional collaborative filtering algorithms, deep neural network architectures, and the user interest vectors derived from the large language model. This hybrid strategy combines the semantic comprehension capabilities of large language models with the predictive power of conventional recommendation techniques, improving both recommendation accuracy and personalization levels.

The recommendation justification module utilizes the large language model to automatically generate natural language explanations for each recommended item. These explanations clarify why specific products are suggested, increasing the interpretability of the recommendation process, fostering user trust, and mitigating the risk of the information cocoon effect.

4.3 System Implementation and Deployment Considerations

Key considerations for system implementation and deployment encompass computing resource allocation, inference latency optimization, and data privacy protection. Efficient distribution of computational resources ensures scalability under varying workloads. Optimizing inference latency guarantees responsive user experiences, particularly in real-time recommendation scenarios. Meanwhile, robust data privacy safeguards ensure compliance with regulations and protect sensitive user information.

Together, these measures underpin the reliability, efficiency, and sustainability of the deployed recommendation system.

5 Conclusion

Traditional recommendation systems excel at mitigating information overload but fall short in understanding deep user needs, addressing cold starts, recommending long-tail products, breaking information cocoons, and providing explainable recommendations. LLMs offer new solutions by enhancing semantic understanding and reasoning, improving modeling for cold-start scenarios, and surfacing long-tail product associations. Experiments demonstrate the proposed LLM-integrated system's superiority in accuracy, recall, and cold-start/long-tail scenarios. Challenges remain, including high compute costs, latency, and privacy concerns. Future research should focus on lightweight/real-time optimization, privacy-preserving techniques, domain adaptation, multimodal modeling, and enhanced explainability.

References

1. MICHALOWSKI M. How much data is generated every day in 2025[EB/OL]. (2025-07 10)[2025-08-08]. https://spacelift.io/blog/how much-data-is-generated-every-day.
2. LIU Q, ZHAO X, WANG Y, et al. Large language model enhanced recommender systems: Taxonomy, trend, application and future[J]. arXiv:2412.13432, 2024.
3. MINAEE S, MIKOLOV T, NIKZAD N, et Large language models: a survey [J]. arXiv:2402.06196, 2024.
4. MIN B, ROSS H, SULEM E, et al. Recent advances in natural language processing via large pre-trained language models: A survey [J]. ACM Computing Surveys, 2023, 56(2): 1-40.
5. Petruzzelli A, Musto C, Laraspata L, et al. Instructing and prompting large language models for explainable cross domain recommendations. Proceedings of the 18th ACM Conference on Recommender Systems. Bari: ACM 2024. 298–308. [doi: 10.1145/3640457.3688137]
6. Lu HY, Ma WZ, Wang YF, et al. User perception of recommendation explanation: Are your explanations what users need? ACM Transactions on Information Systems,2023, 41(2): 48. [doi: 10.1145/3565480