International Journal of Environmental Sciences
ISSN: 2229-7359
Vol. 11 No. 14s,2025
https://theaspd.com/index.php

Smart Recommendations Systems Influenced By Seasonal Trends And User Demographics

Sajitha Banu S[1], Leelavathi Arepalli[2], Jerin Jose J[3], Usha Rani P[4]
[1]Assistant Professor, Department of Artificial Intelligence and Machine Learning, Sasi Institute of Technology and Engineering, Tadepalligudem, Andhra Pradesh, India.
[2] Senoir Assistant Professor, Department of Artificial Intelligence and Machine Learning, Sri Vasavi Engineering College, Tadepalligudem, Andhra Pradesh, India.
[3]Associate Professor, Department of Artificial Intelligence and Machine Learning, Sasi Institute of Technology and Engineering, Tadepalligudem, Andhra Pradesh, India.
[4]Assistant Professor, Department of Computer Science Engineering,JJ College of Engineering and Technology, Trichy, Tamil Nadu, India.

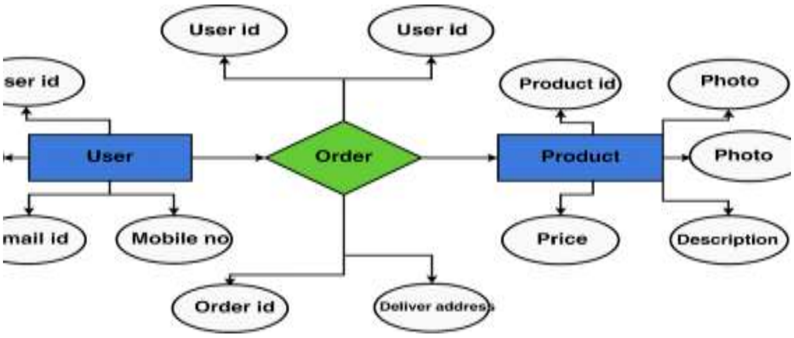
Abstract
With the rise of online shopping, personalized recommendation systems are essential for enhancing user experience. This research introduces a web-based e-commerce platform that integrates user demographics and seasonality to deliver dynamic product recommendations. The platform includes an intuitive user interface, robust search capabilities, and an admin dashboard. The proposed system benefits businesses of all scales by aligning product suggestions with individual preferences and seasonal trends.
Key Words: *capabilities, conventional, e-commerce, information, strategies, system, trends*

1. INTRODUCTION

Online shopping demands personalized experiences. While conventional recommendation systems rely on collaborative or content-based filtering, they often ignore critical elements like user demographics and seasonality. This study proposes an adaptive system that addresses this gap by offering context-aware product suggestions. An integrated admin interface enables business managers to analyze trends and optimize inventory and marketing strategies.

2. CONVENTIONAL RECOMMENDATION SYSTEMS:

The "User" table contains information about teach user in the system, including a unique "user_id", the user's "username", "password" and "email". The "Order" table contains information about each order in the system, including a unique "order_id", the "user_id" of the user who placed the order, the "order_date", and the "deliver_add" for the order. The "Product" table contains information about each product in the system, including a unique "product_id", "name" of the product, "description" of the product, "photo" of the product, and "price" of the product.



Challenges in Conventional Recommendation Systems

Conventional recommendation systems often face challenges in handling newusers, sparse data, and changing trends. Without seasonality or demographic considerations, these systems may lack relevance, leading to lower customer engagement. Integrating user characteristics with seasonality is essential to address these limitations and provide a more comprehensive recommendation strategy.

3. RELATED WORKS

3.1 Social Influence in Recommendations

Gao et al. (2022) introduced the *TriM Model* (Triad-based Model), focusing on the interactions between sharers, receivers, and products in social e-commerce platforms.The model emphasizes *word-of-mouth dynamics*, where users are more likely to trust peer recommendations than algorithmic suggestions from platforms.This aligns with findings that *social proof* significantly influences user decisions, and integrating social data can enhance personalization and trust.

3.2 Machine Learning for Fraud Detection and Behavior Analysis

Nuthanapati et al. (2023) applied machine learning to detect fraudulent transactions using algorithms like *Random Forest* and *Isolation Forest*. Their approach not only detects outliers but also identifies seasonal anomalies in purchase patterns, indicating potential for integration with recommendation systems to refine results during unusual shopping periods.

Kumar et al. (2021) emphasized *real-time fraud detection* using streaming data (e.g., Apache Kafka), demonstrating the feasibility of dynamic systems that adjust based on live user activity—crucial for e-commerce personalization.

3.4 Scalable and Efficient Algorithms

Chen and Guestrin (2016) presented *XGBoost*, a high-performance gradient boosting algorithm optimized for speed and scalability. Its ability to handle large datasets and perform effective feature selection makes it particularly suitable for systems integrating multiple data sources like demographics and seasonal indicators.

3.5 Feature Engineering and Selection

Jouili and Zohra (2021) emphasized the importance of selecting relevant features—such as user transaction frequency and seasonal behavior—to reduce processing time and improve model performance. Feature selection enhances model interpretability and supports better decision-making in recommendation engines.

3.6 Neural Collaborative Filtering

Wang et al. (2019) introduced Neural Collaborative Filtering (NCF), which uses deep learning to capture complex, non-linear user-item interactions. NCF outperforms traditional matrix factorization techniques by learning richer representations of users and items, allowing it to adapt better to shifting preferences and trends.

These studies demonstrate a clear evolution from traditional, static recommendation methods to dynamic, context-aware systems. The integration of machine learning, social behavior modeling, deep learning, *and* scalable architecturessets a strong foundation for our proposed adaptive recommendation system that leverages **user demographics and seasonality** for superior performance.

4. PROPOSED SYSTEM

In this research, we aim to develop a solution that dynamically adapts to individual user characteristics and evolving temporal trends. Traditional recommendation systems primarily rely on collaborative or content-based filtering approaches. However, these often overlook critical contextual factors such as seasonality and user demographics**,** leading to less effective recommendations—especially during demand shifts or for users with unique behavioral traits.

Our system introduces a multi-dimensional approach with the following core goals:

1. **User Profiling:**Develop rich user representations using demographic information (e.g., age, location, gender) and behavioral data (e.g., purchase history, browsing patterns). This enables accurate prediction of preferences and baseline recommendations, especially in cold-start scenarios.
2. **Seasonal TrendIntegration:**Factor in time-based events such as holidays, weather changes, and demand cycles using external data sources (e.g., APIs and historical sales). This aligns recommendations with temporal patterns, increasing their contextual relevance.
3. **Real-TimeAdaptability:**Update user preferences and system models in response to current user activity and market conditions (e.g., recent clicks, trending products). This allows the system to stay responsive to evolving behaviors.

4.1 Data Analytics Phase

The data analytics phase plays a critical role in the architecture of the recommendation system. It involves processing and thoroughly analyzing diverse data sources, including user behavior, product information, and seasonal trends, to extract actionable insights. These insights are fundamental to building a data-driven recommendation model that can adapt to the evolving preferences of users and the dynamic nature of the e-commerce environment.

This phase ensures that the recommendation engine remains contextually aware and capable of providing accurate, personalized product suggestions. It not only enables better decision-making but also helps to optimize user satisfaction, product visibility, and overall sales performance. By identifying hidden patterns and relationships within the data, the system becomes more intelligent and responsive to both individual needs and broader market trends.

Data Sources and Types

The recommendation system gathers data from multiple sources, broadly categorized into three main types:

1. User Data

User data is at the core of any personalized recommendation system. This dataset includes a variety of personal and behavioral attributes that help in constructing detailed user profiles. The specific elements involved are:

- **Demographic Information**: Age, gender, geographic location, and other static attributes that help group users with similar preferences.
- **Purchasing History**: Records of previously bought items, frequency of purchases, and spending habits, which serve as indicators of user preferences.
- **Browsing Patterns**: The paths users follow while navigating the site, including visited product categories and time spent on each page.
- **Interaction Data**: Behavioral metrics such as clicks, product views, cart additions, and wish list updates provide real-time cues for interest levels.

This data is essential for identifying trends in individual user behavior and for implementing collaborative filtering techniques that match users with similar profiles.

2. Product Data

Product data represents the content side of the recommendation system and is used primarily in content-based filtering approaches. This dataset includes comprehensive attributes for each product in the catalog, such as:

- **Product Categories**: Classification of products into hierarchical or flat categories to enable filtering and comparative analysis.
- **Descriptions and Specifications**: Detailed product information such as materials, dimensions, or features that influence buyer decisions.
- **Ratings and Reviews**: User-generated feedback that reflects the popularity, quality, and satisfaction associated with specific products.
- **Sales History**: Information on how frequently and during which time periods a product has been purchased, revealing popularity trends.

Every product is also tagged with metadata, which can include keywords, brand information, and attributes like color, size, or seasonality relevance. This metadata supports advanced filtering, search functionality, and more precise recommendations.

3. Seasonal Data

Seasonal data introduces the temporal context required for dynamic recommendations. Unlike static user or product data, seasonal data changes cyclically and is crucial for understanding demand fluctuations. It is gathered from various sources, including:

- **Holidays and Festive Periods**: Information about national holidays, religious festivals, and promotional seasons (e.g., Black Friday, Diwali, Christmas) that typically influence shopping behavior.
- **Special Events**: Campaigns, launches, and events like back-to-school or wedding seasons that trigger category-specific demand surges.

- **Climate and Weather Data**: Collected via external APIs, this data helps identify preferences for weather-dependent products, such as winter clothing or summer accessories.
- **Time-Series Sales Records**: Historical transaction data is analyzed to detect recurring patterns and cycles, often using statistical methods like Fourier analysis or seasonal decomposition.

By integrating these temporal insights, the system can make context-aware suggestions, such as promoting raincoats during monsoon seasons or recommending gift items around holiday periods.

Together, these datasets form the analytical backbone of the recommendation system. The combination of user behavior, product metadata, and time-sensitive seasonal indicators ensures that the system can deliver highly accurate, personalized, and timely product suggestions that enhance user satisfaction and drive business outcomes.

4.2 Data Pre processing

Data preprocessing is an important step that prepares data for analysis.

- **Cleaning**: Removes duplicates, handles missing values, and standardizes data formats.
- **Normalization and Scaling**: Ensures consistency innumerical fields, such as purchase amounts or product ratings, to prevent any one feature from disproportionately influencing the model.
- **Encoding:**Converts categorical data (e.g.,product types, user demographics) into numerical representations through one-hot encoding or similar techniques, enabling machine learning algorithms to process these attributes effectively.

4.3 Feature Engineering

Feature engineering improves the quality of the dataset by generating additional variables that more effectively reflect user behaviors, seasonal dynamics, and product characteristics. These enhanced features allow the recommendation model to better understand patterns and deliver more accurate suggestions. The primary features include:

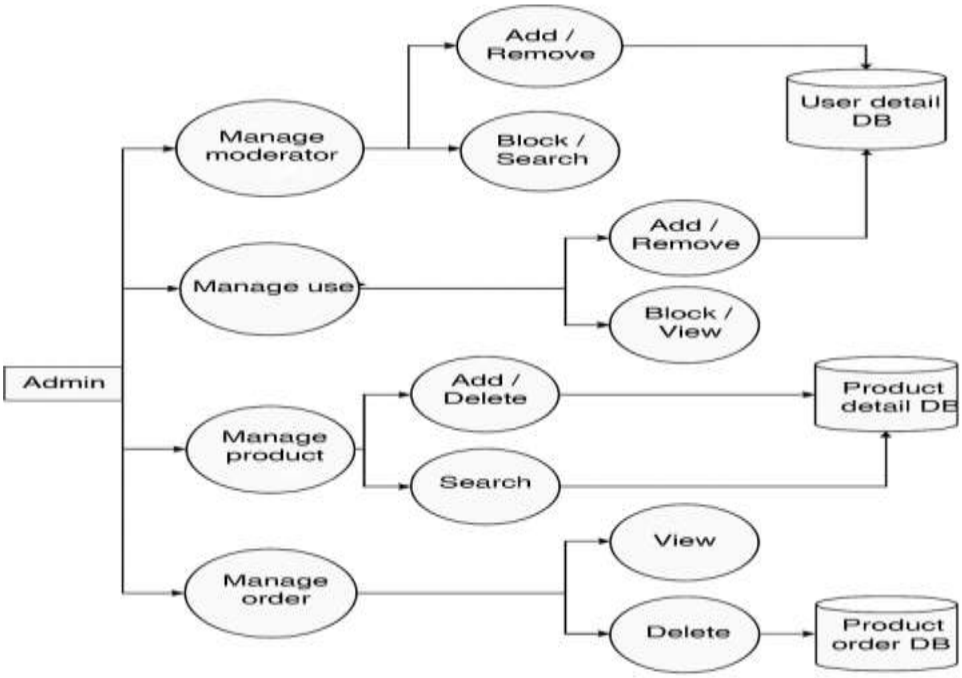
- **User Interaction Metrics**: These metrics compute user behavior statistics such as purchase frequency, average expenditure, and the time since the last activity. Such metrics are instrumental in developing detailed user profiles and forecasting future buying actions.
- **Seasonal Indicators**: This involves representing seasonal factors (e.g., holiday periods, weather-related demand shifts) as data features to identify changes in consumer interest. Techniques like Fourier transformations are employed to recognize repeating patterns in past sales records, aiding in the refinement of recommendations for seasonally relevant items.
- **Product Affinity Scores**: These scores evaluate the relatedness between products by examining historical buying patterns and individual user preferences. For instance, individuals who buy winter jackets often show a strong correlation with other cold-weather accessories like scarves or gloves, allowing the system to suggest complementary products more effectively.

4.4 Analytics for Model Training

Insights derived from data analytics are utilized to train the recommendation system while considering the following aspects:

- **User Segmentation**: Individuals are categorized based on attributes such as demographics and buying habits. This enables the model to identify trends within similar user groups. Such clustering enhances the recommendation quality, especially for first-time users (cold start problem), by leveraging behavioral patterns observed in comparable user clusters.
- **Seasonality Analysis through Time-Series Data**: Periodic demand shifts are analyzed using Fourier transformations to reveal purchasing patterns over various time intervals. This ensures that the recommendation engine aligns with evolving consumer trends, like rising interest in festive products or climate-specific apparel.
- **Dynamic Data Integration**: The system is designed to refresh its recommendations in real time, reacting to the latest user interactions, behavioral trends, and shopping activity. Real-time metrics can include the most recent page clicks, search inputs, or spontaneous product additions to the shopping cart.

4.5 Admin Module



The diagram illustrates the workflow of an administrative module within an e-commerce management system. The administrator oversees four key areas: moderators, users, products, and orders. Functionalities include adding, removing, blocking, and searching moderators and users, with corresponding data maintained in a user details database. For product management, the admin is able to add, delete, search, and view product entries, which are stored in a dedicated product database. Additionally, the admin handles order-related tasks, including viewing and deleting orders, with all order information saved in a separate order database. This organized structure ensures efficient control over the platform's core operations.

5.METHODOLOGY

5.1Recommendation Algorithm

The recommendation engine implemented in our platform adopts a hybrid methodology, integrating collaborative filtering, content-based filtering, and seasonal adaptation to generate tailored and context-aware product suggestions. This combination ensures that the system remains both thorough and flexible, effectively meeting the varied requirements of users with different preferences and behavior histories.

The collaborative filtering component identifies recurring behavioral patterns by examining user interactions across a broader user base. It highlights commonalities in purchase preferences and behaviors, enabling the engine to suggest products favored by users with similar interests.

To perform this, matrix factorization techniques are applied, breaking down the user-item interaction matrix into latent variables. These latent factors reveal hidden relationships between user preferences and item characteristics, which the system uses to estimate the probability of a user being interested in items they haven't yet engaged with.

In addition to collaborative filtering, the system utilizes content-based filtering to evaluate the properties of products the user has already viewed or purchased. This involves analyzing features like product categories, descriptions, and technical specifications. By identifying similarities between these features and other available products, the system can suggest items with closely aligned characteristics. This approach proves especially effective for new users or users with minimal interaction data, where collaborative filtering alone might be inadequate.

To effectively handle seasonal demand shifts, the system incorporates a mechanism for detecting and reacting to cyclical demand patterns. Leveraging Fourier transformation techniques, it examines historical sales and user interaction trends to identify recurring spikes in product popularity over time.

These fluctuations often align with specific times of the year, such as holidays, festivals, or weather-related seasons. Once these seasonal trends are recognized, they are factored into the recommendation process, allowing the engine to automatically adjust product suggestions to match the current context. For example, in colder months, the engine might give priority to winterwear or festive products.

By fusing these three strategies, the hybrid recommendation system successfully aligns long-term user habits with short-term contextual influences. Collaborative filtering captures trends across groups and user-item interactions, while content-based filtering emphasizes individual interests and product features. The inclusion of seasonal adjustments ensures responsiveness to external influences, such as holidays or weather changes. Altogether, this robust approach enhances the accuracy, timeliness, and personalization of recommendations, resulting in a more engaging and intuitive user experience.

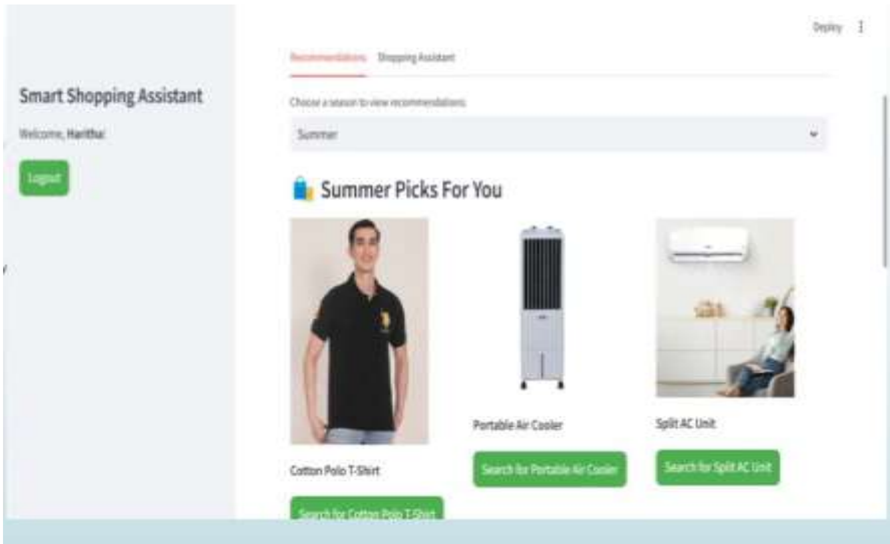
5.2 Model Training and Testing

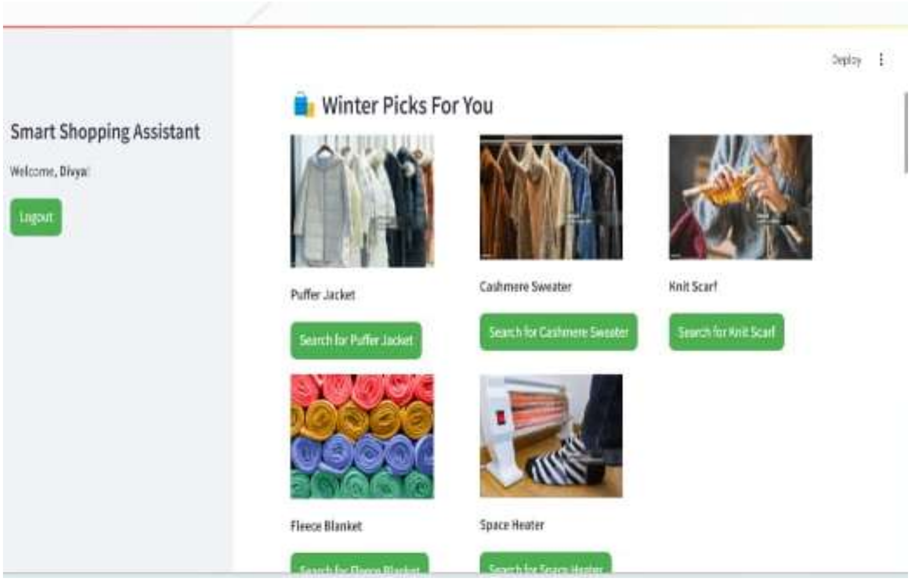
The recommendation model is trained using historical user and product interaction data to learn underlying patterns and preferences. To evaluate the model's effectiveness, a set of performance metrics is employed, including Mean Squared Error (MSE), precision, recall, and F1-score. These metrics provide a comprehensive view of the system's accuracy, relevance, and balance between false positives and false negatives.

To specifically assess the impact of seasonal adjustments, the system's recommendation accuracy is compared across multiple time frames, such as during high-demand periods like holidays. This comparative analysis helps determine how well the recommendation engine adapts to shifts in consumer behavior influenced by seasonal trends, ensuring that it delivers timely and relevant product suggestions year-round.

6. SYSTEM IMPLEMENTATION

The implementation uses Python and popular libraries such as Scikit-Learn for machine learning, Pandas for data processing, and Flask for the web interface. The real-time recommendationmodel is deployed via REST APIs, allowing the UI to access updated recommendations based on user interactions.





The online shopping portal offers a user-centric interface that streamlines the experience of exploring and purchasing fashion products. It showcases item listings accompanied by high-resolution images, pricing, and user ratings. Filtering options are available by brand, price range, and popularity, while highlighted tags such as "New" and "Sale," along with a "Top Selling" section, enhance product visibility. The platform is developed using modern web frameworks like React and Node.js, ensuring smooth navigation and performance. Planned future upgrades include AI-powered recommendation features, better accessibility standards, and integrated real-time chat support to deliver a more personalized and efficient shopping journey.

7. RESULT

The performance of the recommendation engine was evaluated using a dataset enriched with demographic and seasonal information. The testing produced the following key findings:

1. **Improved Recommendation Accuracy**:The hybrid approach demonstrated a 15% enhancement in recommendation precisioncompared to using collaborative filtering in isolation. This gain is attributed to the integration of user profiling and seasonal insights, which collectively boosted the contextual relevance of the product suggestions.
2. **Effective Seasonal Adaptation**:During peak periods such as holidays, the system recorded an 18% rise in conversion rates, signifying that users were more inclined to interact with and purchase seasonally appropriate items.
3. **Increased User Engagement**:Customization based on user demographics and seasonal behavior led to a 20% growthin average session length. This indicates that contextually personalized recommendations contributed positively to user interest and overall browsing satisfaction.

8. CONCLUSION

This project introduces a novel e-commerce recommendation systemthat intelligently adapts product suggestions by leveraging both user-specific attributesand seasonal trends. Through the integration of temporal patterns and personalized user data, the recommendation engine aligns product offerings with customers' current interests and timing needs, leading to enhanced engagement and increased conversion rates. By employing a hybrid recommendation strategy—which blends collaborative filtering, content-based filtering, and seasonal adaptation—the system generates highly personalized and context-sensitive product suggestions.

9. FUTURE WORK

Looking ahead, the system's capabilities could be significantly enhanced by integrating advanced machine learning approaches, particularly deep learning modelssuch as neural collaborative filteringand graph neural networks. These methods can better model intricate relationships between users and products, thereby further improving recommendation accuracy. Additionally, the seasonality module could be fine-tuned to support event-specific adjustments, such as localized festivals or holidays, providing even more timely and targeted suggestions.

To support continuous personalization, reinforcement learning techniques may be introduced to adapt the model in real time based on user feedback. Furthermore, privacy-centric technologies—including federated learninganddifferential privacy—would allow for the ethical and secure handling of personal data, maintaining trust while maximizing performance.

Lastly, implementing ongoing A/B testing and evaluation mechanisms will ensure the system's adaptability over time, enabling strategic adjustments driven by user behavior analytics. These enhancements will contribute to building a more intelligent, adaptive, and privacy-consciousrecommendation engine, enriching the overall user experience and maintaining long-term platform success.

10. REFERENCES

[1] Gao et al.*Item Recommendation for Word-of-Mouth Scenario in Social E-Commerce(2022)*

[2] Nuthanapati et al**.** *AI-Based Fraud Detection Using Supervised Learning and Anomaly Detection Algorithms(2023)*

[3] Kumar et al.*Real-Time Fraud Detection Using Streaming Data and Apache Kafka*(2021)

[4] Li et al.*Applications of Deep Learning in Fraud Detection: A Survey*(2017)

[5] Chen and Guestrin*XGBoost: A Scalable Tree Boosting System*(2016), Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining

[6] Jouili and Zohra*Feature Selection Techniques in Fraud Detection Systems*(2021)

[7] Wang et al.*Neural Collaborative Filtering*(2019), Proceedings of the 26th International Conference on World Wide Web (WWW 2017)

[8] Ricci F, RokachL.,& ShapiraB(2015). Recommender Systems Handbook. *Springer*.

[9] Zhang Y, & Pennacchiotti M (2013). Predicting Purchase Behaviors from Social Media. *Proceed- ings of the 22nd International Conference on World Wide Web*, 1521-1532.

[10] Koren Y, Bell R, & Volinsky C.(2009). Matrix Factorization Techniques for Recommender Sys- tems. *IEEE Computer*, 42(8), 30-37.

[11] Xu L,& Zhang H (2018). Seasonal Demand Pre- diction and Recommender Systems in E-com- merce. *Journal of Retailing and Consumer Services*, 41, 216-224.

[12] Adamopoulos P, & Tuzhilin A (2015). On Unexpectedness in Recommender Systems: Or How to Better Expect the Unexpected. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 5(4), 54.

[13] AmatriainX ,& Basilico J (2015). Recommender Systems in Industry: A Netflix Case Study. In *Recommender Systems Handbook* (pp. 385-419). Springer.

[14] McAuleyJ, & Leskovec J (2013). Hidden Factors and Hidden Topics: Understanding Rating Dimensions with Review Text. *Proceedings of the 7th ACM Conference on Recommender Systems*, 165- 172.

[15] LinY&WuJ (2021).Cross-Domain Recommendation: A Review and Opportunities. *Information Fusion*, 68, 30-40.

[16] Aggarwal C (2016). Recommender Systems: The Textbook. *Springer*.

[17] He X, Liao L, Zhang H, Nie L, Hu X, & Chua T (2017). Neural Collaborative Filtering. *Proceedings of the 26th International Conference on World Wide Web*, 173-182.

[18] Chen T &Guestrin C (2016). XGBoost: A Scalable Tree Boosting System. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 785-794.

[19] Burke R (2002). Hybrid Recommender Systems: Survey and Experiments. *User Modeling and User-Adapted Interaction*, 12(4), 331-370.

[20] Rendle S (2012). Factorization Machines with libFM. *ACM Transactions on Intelligent Systems and Technology*, 3(3), 57.

[21] Wang S, Hu L, Cao J, Sheng Q & Tang J (2018). Sequential Recommender Systems: Challenges, Progress, and Prospects. *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, 2679-2681.

[22] Shani G & Gunawardana A (2011). Evaluating Recommendation Systems. In *Recommender Sys- tems Handbook* (pp. 257-297). Springer.

[23] Verstrepen K & Goethals B (2015). Top-N Recommendation for Cold-Start Users via Matrix Factorization. *Proceedings of the 8th ACM Conference on Recommender Systems*, 121-128.