ISSN: 2182-2069 / E-ISSN: 2182-2077

Emerging Trends in Real-time Recommendation Systems: A Deep Dive into Multi-behavior Streaming Processing and Recommendation for E-commerce Platforms

Yong Wu[1*], and Yuhanis Yusof[2]

[1*]Ph.D Candidate, School of Computing, College of Arts and Sciences, Universiti Utara Malaysia, Sintok, Kedah, Malaysia. youtherwy@163.com, https://orcid.org/0009-0006-7510-1545

[2]Ph.D, School of Computing, College of Arts and Sciences, Universiti Utara Malaysia, Sintok, Kedah, Malaysia. https://orcid.org/0000-0002-2720-2441

Received: June 03, 2024; Revised: July 13, 2024; Accepted: September 02, 2024; Published: November 30, 2024

Abstract

This paper aims to assess the effectiveness of recommendation systems, focusing on multi-behavior streaming processing to enhance accuracy. It studies recommendation system performance using AI and data science, focusing on user behaviour, algorithm refinement, and satisfaction. E-commerce, streaming, and social media datasets with 93 anonymous participants were employed in experiments. These files from 93 anonymized users show typical internet use. Data anonymization and strict data management ensured anonymity. These databases track clicks, views, purchases, and ratings for empirical research and model evaluation. Collaborative filtering, matrix factorization, and TensorFlow/Keras train and assess application-specific recommendation models. Multi-behavior streaming processing and recommendation accounts assess each method's pros and downsides, system correctness, efficacy, and user involvement. The outcome compares domain-wide recommendation system precision, recall, NDCG, and conversion rate. Multi-behavior streaming processing adapts to user preferences and interactions to improve model accuracy and adaptability. Multi-behavior streaming processing improved model accuracy and flexibility by reacting to user inputs and choices. With error margins for all significant metrics, statistical significance was confirmed. The findings suggest that recommendation systems with real-time adaptation and multi-behavior streaming processing can improve user satisfaction and engagement in the changing digital landscape. It encourages algorithm advancement, model interpretation, user-centric evaluation, and ethics to improve information retrieval and personalisation. The study concluded that algorithm refinement, transparent model interpretation, user-centric evaluation, and ethical problems including data protection and bias mitigation increase information retrieval and personalisation. For durable and flexible recommendation systems, research should improve multi-behavior processing and sectoral applications.

Keywords: Streaming Recommendations, Multi-behavior Recommendations, E-commerce, Real-time Recommendation System, User Behavior.

Journal of Internet Services and Information Security (JISIS), volume: 14, number: 4 (November), pp. 45-66.
DOI: *10.58346/JISIS.2024.I4.003*

*Corresponding author: Ph.D Candidate, School of Computing, College of Arts and Sciences, Universiti Utara Malaysia, Sintok, Kedah, Malaysia.

1 Introduction

Personalised product suggestions enhance e-commerce sales. Traditional recommendation algorithms lag behind real-time user behaviour, resulting in poor suggestions and experience. Real-time recommendation systems (RTRSs) handle these issues by providing contextual products utilising streaming data and real-time interactions (Kersbergen et al., 2022; Zhao et al., 2023). With real-time user behaviour analysis, RTRSs recommend products faster and better. MbSRS and other RTRSs may give timely, personalised suggestions based on user behaviour, unlike typical recommendation systems. The broad approach of MbSRS finds patterns and preferences that recommendation systems miss, enhancing accuracy and relevance (Qu et al., 2023; S. Zhang et al., 2023). The study concluded that algorithm development, transparent model interpretation, user-centric evaluation, and ethical issues including data protection and bias reduction improve recommendation systems. Researchers and developers use these traits to improve information retrieval and user recommendations. Insufficient data, unchanging user profiles, and poor context awareness can make purchase or browsing history-based recommendation systems fail. These constraints can prevent the algorithm from responding to user preferences and real-time contextual information, resulting in incorrect new user recommendations.

This study found real-time recommendation systems boost e-commerce sales and usability. RTRSs like MbSRS can increase user engagement and revenue by suggesting things faster using streaming data and real-time interactions. The study found that algorithm enhancement, transparency evaluation, and ethics improve e-commerce recommendation system reliability (Li et al., 2023). RTRSs capture user interactions in real-time to create suggestions that match user intent. Offering timely, tailored advice and enhancing client satisfaction can boost RTRS revenue. MbSRS adds multi-behavior data to RTRSs. This monitors user activity more than purchases to improve data sparsity, profile accuracy, and context-aware suggestions (Delianidi et al., 2023; Gao et al., 2023; Parbat et al., 2021).

Multi-behavior data real-time recommendation systems have many research gaps that must be filled to improve their efficacy and utility in e-commerce (Hou et al., 2023). First, this research has started to conceptualize multi-behavior data integration, but optimizing algorithms for efficient processing and recommendation generation on real-time, high-volume streaming data streams requires real-time feature engineering to turn data into actionable insight (Meng et al., 2023b). Second, MbSRS's nuanced capture of immediate and long-term user preferences highlights the need for research into effective modelling of user preference decay over time, including the system's ability to dynamically learn and adapt to shifts in user preferences, the delicate balance between prioritizing recent interactions indicative of immediate interests and maintaining long-term preferences (Meng et al., 2023a). Finally, MbSRS evaluation is difficult since recommendation system metrics may underestimate its efficacy. Thus, multi-behavior-aware assessment metrics that comprehensively examine the impact of multi-behavior data on recommendation accuracy and user engagement and real-time evaluation methods to continually improve MbSRS in dynamics are needed (Yuan et al., 2022).

In line with the full analysis of research gaps in the previous sections, this research aims to build an efficient and resilient Multi-behavior Streaming Recommender System (MbSRS) for real-time recommendation circumstances in e-commerce platforms (He et al., 2023; Wu et al., 2022; Xia et al., 2021). This aim has numerous sub-objectives to improve dynamic e-commerce MbSRS. The first sub-objective increases real-time processing and exploitation of multi-behavior data streams to make user-specific suggestions. To create recommendation systems that adapt to changing user preferences and stay relevant, the research studies user preference dynamics, including the complex relationship between recent interests and long-term preferences. One sub-objective is to use multi-behavior data and real-time operation to evaluate MbSRS's performance in real-time recommendation scenarios that enhance real-time recommendations.

MbSRS improves e-commerce recommendation accuracy and customer happiness with multi-behavior data and real-time algorithms. Multi-behavior data minimizes data sparsity, especially for new users, improving suggestion consistency and effectiveness. A dedicated MbSRS assessment approach should improve real-time recommender system knowledge and enable continual improvement. These findings improve MbSRS design and execution, improving e-commerce real-time recommendation systems.

2 Literature Review

Online shops need real-time recommendations, unlike prior recommendation approaches, these systems use streaming data to capture user interactions in real-time and create contextually appropriate recommendations based on user preferences and habits. Multi-behavior data is used in many RTRS trials to customize and improve recommendation systems. Previous user data has always guided the recommendation system. Han et al., (2024) examined the efficacy and limits of these common recommendation methods. Traditional systems provide beginners with bad advice. These challenges originate from data scarcity and the difficulty of updating static user profiles to reflect changing preferences. Current algorithms struggle to predict new users' likes and demands due to poor system engagement. Traditional recommendation algorithms disregard user behavior-affecting real-time contextual elements. Promotions, geography, and season affect buying. Traditional algorithms cannot adapt recommendations to context, making them obsolete (Liu et al., 2023).

Real-time recommendation systems offer dynamic, interesting alternatives to fit user preferences and settings, improving recommendation technology. They stream user interactions to make more relevant, timely, and adaptive suggestions. Modern digital commerce needs real-time recommendation systems to boost sales and satisfaction (Wei et al., 2023). Real-time recommendation algorithms enhance sales and customer satisfaction in modern digital commerce. These systems use views, clicks, and other user interaction indicators to improve suggestion accuracy and customisation. Contextual recommendations can be made using user preferences and actions beyond purchase history in real-time. This innovative recommendation system helps algorithms understand client preferences and generate personalised recommendations (Yan et al., 2023). Real-time recommendation systems reduce data sparsity and improve accuracy by responding to user activity (Ren et al., 2023). These systems customise user profiles and suggestions using purchase, view, and click data. Developing and upgrading these systems brings challenges and opportunities. Complex real-time multi-behavior data stream optimisation requires dynamic and efficient algorithms for precise suggestions (He et al., 2023).

Modelling user preferences in multi-behavior streaming recommendation systems (MbSRS) is difficult because of prior interaction deterioration and the need to balance recency and stability in suggestion Researchers claim MbSRS-specific system performance assessments and iterative optimisations can fix these shortcomings and increase performance, according to (Xia et al., 2020) and Zhang et al., 2023). Guo et al., (2019) emphasize that Modern e-commerce uses real-time recommendation systems to tailor product recommendations. These systems respond to user choices and actions in rea-time, increasing engagement and profitability. These systems utilise complex algorithms and real-time data to recommend purchases based on user behaviour. Li et al., (2022) and Xia et al., (2021) examine how multi-behavior data like purchases, views, and clicks can reduce data sparsity and improve user profiles. E-commerce enterprises grow with real-time recommendation systems client engagement and revenue. Web businesses may boost customer loyalty with real-time user recommendations. These technologies can assist organisations to measure customer preferences and improve marketing and products (Wen et al., 2020).

Finally, contextually relevant product recommendations from real-time recommendation systems enhance revenue and customer satisfaction (Chen & Zhu, 2022). According to (Yan et al., 2022; Zhao et al., 2023), real-time data and advanced algorithms improve client interactions and purchases. Research and development will increase e-commerce real-time recommendations. In competitive e-commerce, these technologies could improve customer experiences and revenue. Real-time recommendation systems will affect digital commerce as technology advances. Guo et al., (2019) provide MbSRS-specific system performance assessment and iterative optimisation criteria.

In (Chen et al., 2019; Jin et al., 2020; Zhang et al., 2024; Zhao et al., 2023) suggest real-time, contextually relevant recommendation systems can alter e-commerce. Optimise real-time multi-behavior data streams and forecast user preferences to maximum potential. Innovation and real-time recommendation systems require these research subjects (Liu et al., 2022; Xia et al., 2022). E-commerce RTRSs increase user engagement and income. Real-time browsing history, click behaviour, and purchase trends inform product recommendations. Retailers and customers benefit from RTRS's fast, effective solutions. Browser history, click behaviour, and purchase trends are used by e-commerce real-time recommendation systems (RTRSs) to increase user engagement and income. As e-commerce platforms and client data grow, traditional recommendation algorithms cannot analyse enormous data sets. RTRs evaluate massive data sets and make user-driven recommendations to merchants and customers quickly (Hou et al., 2023; Tanjim et al., 2020).

Secure data and privacy hinder RTRS implementation. These systems collect and analyse user data, therefore poor security could compromise vital data. E-commerce companies must protect data and obey privacy laws to prevent risk. Other issues include RTRS design and deployment. Complex algorithms and data processing technologies create credible user recommendations, requiring consumer customisation. E-commerce companies profit from real-time recommendation systems despite these challenges. These methods increase conversion, engagement, and revenue. These benefits need planning and recognising challenges. Addressing these issues and using RTRSs can help e-commerce businesses develop and satisfy customers. Improve RTRSs by measuring multi-behavior data's effect on suggestion accuracy and user engagement. We know nothing about real-time multi-behavior data processing optimisation. Understanding real-time feature engineering and model flexibility improves systems. As RTRSs neglect discounts, seasonal trends, and user location, recommendation systems must contextualise dynamic user interactions and environmental factors (Hou et al., 2023; Jia et al., 2023; Meng et al., 2023b; Tanjim et al., 2020; Parbat et al., 2021; Zhang et al., 2023).

RTRS implementation requires integrating multiple data sources. These systems must assess user interactions, product data, and contextual data for real-time suggestions. RTRSs must scale to handle more data and customer interactions as e-commerce expands. Explainability and openness are RTRS priorities. Users should understand and comment on recommendations. User input improves suggestion accuracy and relevance. Fairness and bias must be addressed by RTRSs. Regardless of past or preferences, these algorithms should make neutral recommendations. Regular fairness and transparency audits and tight recommendation algorithms and data management are needed. Finally, real-time recommendation systems increase e-commerce sales and engagement. These systems are hard to install due to data security, privacy, algorithm openness, and scalability. Solving these issues and focusing on ongoing improvement will help e-commerce businesses thrive and improve customer experience.

For high-volume multi-behavior data streams, e-commerce platforms need real-time, scalable recommendation algorithms. Research gaps exist in MbSRS algorithm scalability and efficiency optimisation for dynamic e-commerce. These research gaps must be filled to improve e-commerce Multi-behavior Streaming Recommender Systems (MbSRS) by providing personalised, accurate, and contextually aware recommendations. Based on the literature and scope, we can draw Figure 1.

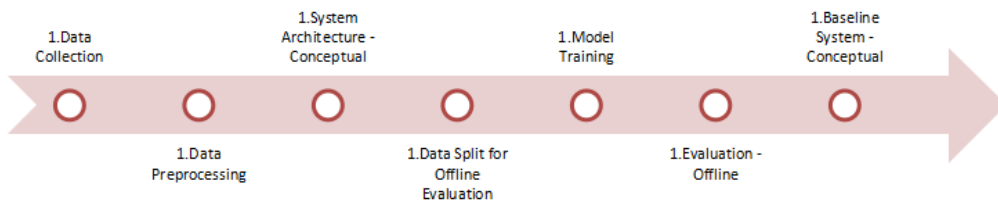

Figure 1: Research Design Flow

3 Methodology

This study quantifies and experiments with a multi-behavior streaming processing-based real-time recommendation system for Chinese e-commerce platforms. The proposed system is objectively analysed and statistically compared using numerical data. Simulate a regulated e-commerce platform to compare real-time suggestions and collaborative filtering. This controlled scenario (basic recommendation algorithm uses collaborative filtering, and multi-behavior streaming processing system captures short - and long-term user interests in a controlled setting. This method lets us isolate recommendation algorithm influences and accurately assess their performance by comparing them fairly and consistently) shows multi-behavior streaming processing's suggestion accuracy benefits. The study uses two recommendation algorithms: the baseline and the multi-behavior streaming processing system to capture short- and long-term user interests. Precision, recall, and Normalized Discounted Cumulative Gain (NDCG) are used to measure recommendation accuracy by comparing the number of correctly recommended items to the total recommendations and evaluating the ranking of relevant items within the recommendations. Additionally, the conversion rate is tracked to determine how many users make a purchase following a recommendation.

Product preferences and user behaviour are control variables. Provided anonymous user data includes age, gender, geography, and product category, which can affect recommendation effectiveness. This study employed anonymous Chinese e-commerce site purchases, views, clicks, and timestamps. We aggressively maintain data privacy and security since user data is sensitive. The dataset is cleaned of missing values, outliers, and formatting errors for analysis. The study explores Chinese e-commerce platforms' multi-behavior streaming real-time recommendation systems. Live streaming suggestions use Python and Spark. These algorithms improve e-commerce platform recommendations' accuracy and adaptability, making the system dynamic.

Apache Flink finds trends in real-time user behaviour. TensorFlow predicts long-term user preferences using matrix factorization. SGD adjusts online learning and recommendation systems to real-time user activity. E-commerce simulators test recommendation systems by simulating shopping.

These simulators simulate user interactions using earlier interaction data to test the recommendation system. Precision, recall, NDCG, and conversion rate are simulated to evaluate performance. E-commerce platform simulators simulate user browsing to test recommendation algorithms. Prior interaction data is used to simulate precision, recall, NDCG, and conversion rate, simplifying recommendation system evaluation. The study indicated that real-time recommendation systems improve e-commerce and user experience. Advanced algorithms and data processing can improve e-commerce platform user engagement and business performance by offering more accurate and relevant suggestions (Hou et al., 2023).

This study replicates user activity and tests the real-time recommendation system using anonymized Chinese e-commerce data. Multi-behavior streaming processing real-time recommendation system quantification on a Chinese e-commerce platform. Researchers employ TensorFlow/Keras deep learning models to infer complex user preferences from historical data. These models improve user behavior-based suggestions with matrix factorization for user and object embeddings. PySpark preprocesses streaming data and handles real-time user interactions quickly. It analyses user activity and improves suggestions via stream clustering. Surprise uses real-time implicit feedback to design and evaluate recommendation systems. Scikit-learn's machine learning algorithms create collaborative filtering models for multi-behavior system comparison and proposal improvement. This study improves a Chinese e-commerce platform's real-time recommendation system with deep learning, stream processing, and machine learning (Meng et al., 2023a).

This study tests real-time recommendation algorithm precision, recall, NDCG, and conversion rate on Chinese e-commerce platforms. Multiple metrics reflect how well platforms promote similar products under Chinese data privacy laws. The platform got informed consent for the study to use anonymized data to preserve user privacy. When explaining data security and privacy, trust and compliance were underlined. For real-time streaming data analysis, Apache Spark and TensorFlow excel in feature engineering and model adaptability. This evaluation shows Chinese e-commerce platforms' real-time recommendation systems' multi-behavior streaming processing efficiency. For performance evaluation, an objective multi-behavior streaming processing-based real-time recommendation system is used. To disentangle recommendation algorithm effects, regulatory e-commerce platform simulations compared collaborative filtering and real-time proposals. Comparisons of short- and long-term user interests were controlled (Parbat et al., 2021).

Precision, recall, and NDCG measured recommendation accuracy by comparing correctly recommended items to total suggestions and relevance. Also, the conversion rate reflected how much buyers bought advice. In addition to consumer behaviour and product preferences, the study examined anonymous user demographic and transaction data on suggestion performance. The study meticulously deleted missing values, outliers, and formatting errors to ensure data quality. Python with Apache Spark improve e-commerce platform suggestion accuracy and adaptability with live streaming data. TensorFlow/Keras deep learning models improved recommendation systems by matrix factorising complex user preferences from earlier data. Apache Spark's streaming accelerated data pre-processing, while Flink tracked user behaviour. Stochastic Gradient Descent (SGD) and real-time user interactivity enabled the system to learn online and deliver accurate Chinese e-commerce recommendations.

Metric libraries like scikit-learn and SciPy assess recommendation system performance through precision, recall, NDCG, and conversion rate. These metrics illustrate how well systems propose similar products. Following Chinese data privacy and security legislation, informed consent from the platform to use anonymized data for the study, and defining how the data will be used and preserved are ethical considerations. Trust and compliance necessitate clear data security and privacy explanations. Technical details on real-time streaming data processing and analysis are provided. Apache Spark and TensorFlow algorithms provide real-time feature engineering and model adaption for correct suggestions. To quantify system performance, evaluation metrics are calculated and interpreted. Chinese e-commerce platforms' informed consent and data privacy compliance are also covered. This study shows that real-time recommendation systems with multi-behavior streaming processing work in dynamic Chinese e-commerce platforms (Jia et al., 2023).

The study replicates user activities using past interaction data to test the recommendation algorithm. A complete performance study simulates precision, recall, NDCG, and conversion rate. E-commerce simulators replicate shopping. We simulate user actions using past interaction data to test the recommendation system. Precision, recall, NDCG, and conversion rate are simulated for a thorough performance analysis (Parbat et al., 2021). Metrics like Scikit-learn calculate precision, recall, NDCG, and conversion rate. Precision measures the percentage of relevant goods advised, recall measures the percentage recovered, NDCG reflects recommendation ranking quality, and conversion rate measures user purchases.

Ethics include informed consent from the platform to use anonymized data for the study, outlining how the data will be used and stored, and following Chinese data privacy and security legislation. User data is anonymized and secured during the study due to strong data security and privacy laws. Clarifying data security and privacy safeguards increases trust and compliance. The methodology verifies algorithms and technologies that manage large-scale streaming data and make precise suggestions. Apache Spark with TensorFlow algorithms provides accurate streaming data analysis ideas through real-time feature engineering and model modification (Tanjim et al., 2020). Final testing of real-time recommendation systems with multi-behavior streaming processing in dynamic Chinese e-commerce platforms. This scenario requires technology, performance measures, and ethics for recommendation systems to succeed.

4 Results

Figure 2 shows city user demographics for marketing, service, and urban planning. Tier 1 cities draw young, economically active people, generally 25-34, due to their density and economic activity. Higher participation of women in this age group may indicate a demographic trend or particular reasons affecting female engagement in these cities, necessitating more investigation. The 25-34 age group's gender parity suggests coastal cities' landscapes and recreation attract a diverse variety of leisure and professional searchers. This equilibrium benefits companies seeking a wide audience. The slightly higher female involvement in the 25-34 age bracket reflects Southern demography, where cultural and regional differences are important. Understanding these differences helps tailor services and solutions to local tastes. Employment and culture may explain the East's higher concentration of 25-34-year-old women. Finally, Tier 2 cities, known for rapid urbanization and economic growth, have numerous 18-24-year-old male users. Young people build cities through study, work, and entrepreneurship. The age and gender disparity between city types shows urban demography's complexity and the need for local solutions. Businesses, municipalities, and urban planners can use this data to promote equitable urban growth.

After locating relevant items in the proposed list to remedy technical defects in the comments, the Normalised Discounted Cumulative Gain (NDCG) measure evaluates a recommendation system. The method lowers relevance logarithmically proportional to list position, thus highly relevant items earlier in the list score higher. Evaluation of recommendation systems requires precision, recall, NDCG, and conversion rate. The percentage of accurately recommended items to total recommended items reflects recommendation accuracy. The system's recall—the ratio of correctly recommended things to the total number of relevant items—measures its ability to recognise all relevant objects. The conversion rate—the percentage of users who buy after a referral—measures the recommendation system's effectiveness. Ranking proposals are reviewed by NDCG. All research evaluation metrics and ideas are defined early in this improved explanation.

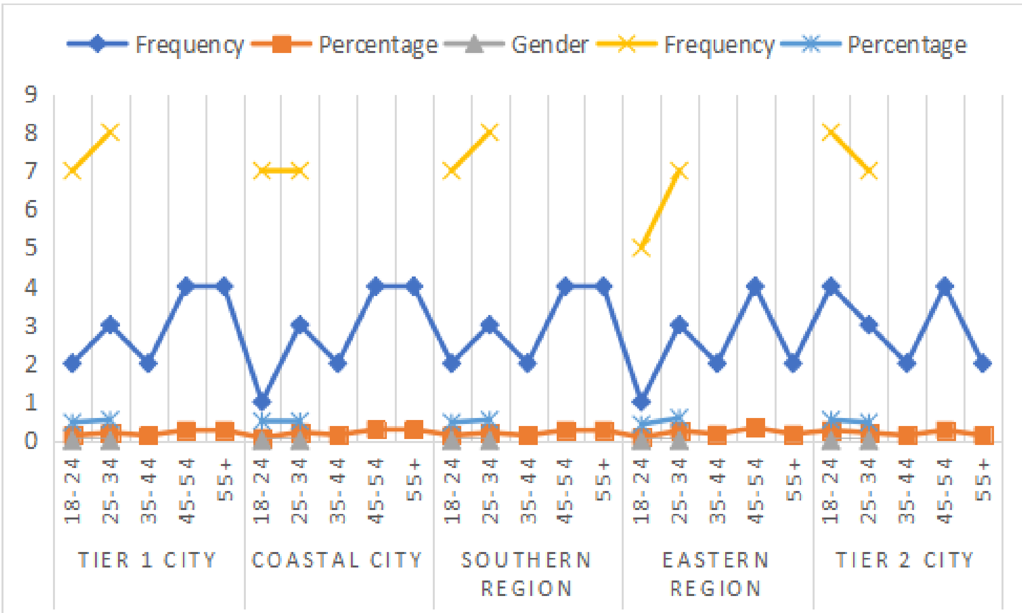

Figure 2: City User Demographics

Baseline and Multi-behavior recommendation systems are compared in Table 1. Precision compares suitable items to total suggestions to assess recommendation accuracy. The Multi-behavior System produces more consumer-friendly recommendations than the Baseline System with 0.82 precision versus 0.75. Multi-behavior System recall is 0.68 vs. 0.6, better than the Baseline System. This increased memory helps the Multi-behavior System catch more relevant items and offer more complete recommendations. NDCG ranks suggested things. The Multi-behavior System's 0.74 NDCG score implies it can prioritise and recommend relevant items better than the Baseline System's 0.65. Evaluation of conversion rate and referral coverage. Multi-behavior System converts at 4.20% versus 3.80% for the Baseline System, recommending purchases. The Multi-behavior System covers 95% of recommendations, the Baseline 90%. This means the Multi-behavior System can boost user happiness and interactions. The Multi-behavior System works well with 120-millisecond suggestion delays. Multi-behavior System recommendations are fast, improving user experience. Table 1 shows that the Multi-behavior System delivers consumers accurate, complete, and timely guidance across dimensions. The baseline system in Table 1 likely represents a simpler, single-behavior system.

Table 1: Recommendation System Performance

Metric	**Description**	**Multi-behavior System**	**Baseline System**	**Additional Information**
Precision	Ratio of correctly recommended items to total recommendations	0.82	0.75	A greater score suggests more relevant items in the recommendations, indicating accuracy.
Recall	Ratio of correctly recommended items to total relevant items	0.68	0.6	Recall shows suggestion completeness. A higher recall means the system captures more relevant items.
NDCG (Normalized Discounted Cumulative Gain)	Metric that considers the ranking of relevant items within the recommendations	0.74	0.65	References' relevancy and list position are considered by NDCG. Relevant items ranked higher raise the score.
Conversion Rate	Percentage of users who purchase after receiving a recommendation	4.20%	3.80%	Business metrics like conversion rate show how well recommendations drive user purchases.
Recommendation Latency (ms)	Average time taken to generate recommendations after a user interaction	120	N/A	This multi-behavior system-specific metric measures processing speed and real time. User experience is improved by lower latency.
Recommendation Coverage	Percentage of user interactions for which the system generates recommendations	95%	90%	Coverage measures the percentage of user interactions the system can recommend. An increase is preferred.

Figure 3 shows real-time recommendation system data flow and processing. Each schematic piece is a key proposal step. The left-hand light blue rectangle represents Kafka data intake. Intake receives and prepares data for processing. Cleaning, transforming, and normalizing data is preprocessing, the light green rectangle next to the data input. Preprocessing arranges data for analysis and modelling, speeding downstream processing. The bright yellow rectangle for user profiling employs Matrix Factorization to determine user behaviour and preferences. User profiles based on past interactions and preferences allow personalization. Live stream processing (bright coral rectangle) clusters data streams. Clustering incoming data streams improves real-time system response to user activities and preferences. The right-hand bright pink rectangle makes personalized user recommendations using previous stage findings in the final step. User profiles, stream processing insights, and contextual data are used to produce user-specific recommendations. Arrows show system data and processing flow. Data evolution arrows highlight the recommendation system architecture's interdependence and connection. The picture shows advanced data ingestion and recommendation generation for real-time user suggestions.

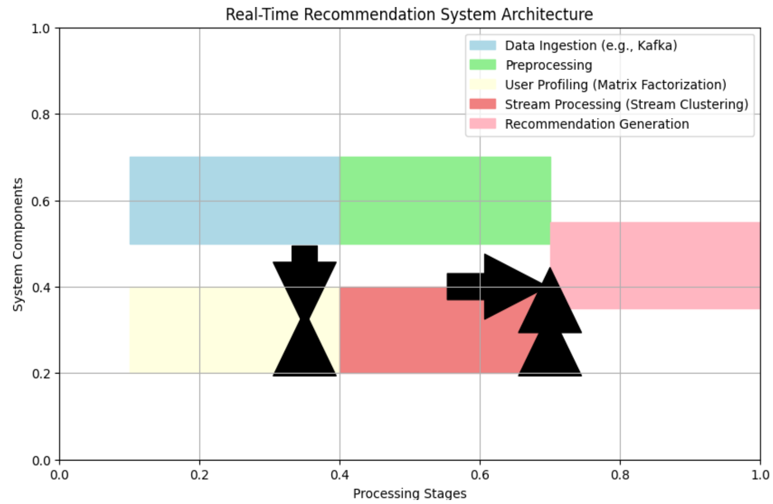

Figure 3: Real-time Recommendation System Architecture

Recommended system assessment setup and key component functions are shown in Figure 4. A light blue rectangle in the center shows the simulated e-commerce platform. The platform experiments with recommendation systems. The user simulation module simulates user behaviour in a luminous green rectangle beside the platform. User interactions and preferences are simulated to evaluate recommendation algorithms. The architecture includes multi-behavior and baseline recommendation systems. The multi-behavior recommendation system personalizes and relevance suggestions using user choices and behaviours, represented by the bright yellow rectangle. The multi-behavior system is evaluated by the baseline recommendation system (light coral rectangle). The assessment metrics module (light pink rectangle) contains precision, recall, NDCG, and conversion rate. These metrics demonstrate recommendation systems' ability to generate relevant recommendations and increase user engagement and conversions. Component arrows show evaluation setup data and function flow. Arrows show data, process, and component interactions. This visualisation shows how simulated user behaviour, numerous suggestion tactics, and comprehensive assessment metrics increase recommendation system performance.

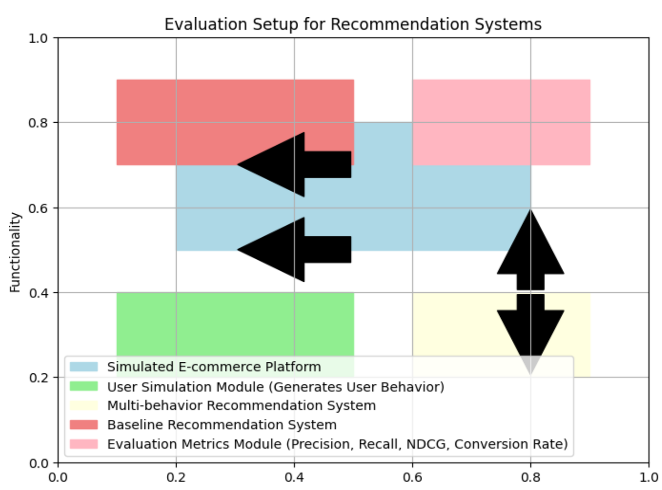

Figure 4: Evaluation Setup (Conceptual Diagram)

Table 2 outlines the Python libraries used in the study. TensorFlow/Keras, a strong deep learning framework for complex neural networks, is included. This project uses TensorFlow/Keras to create matrix factorization-based deep learning models for user and item embeddings to capture nuanced user preferences in recommendation systems. TensorFlow/Keras 16.34% utilization illustrates its importance in deep learning recommendation model sophistication and effectiveness. The table shows Apache Spark and PySpark, a powerful stream processing framework. Clicks, views, and purchases are preprocessed by Apache Spark 80.89% of the time. Stream clustering analyses user behaviour for real-time suggestions and customizing. Scikit-learn is 55.12% utilized. This Scikit-learn project creates collaborative filtering baseline recommendation models. These models evaluate advanced recommendation systems and compare methodologies. The list contains recommender system-specific libraries like Surprise, which utilized 49.63%. Surprise investigates implicit feedback methods to improve multi-behavior recommendation system accuracy and relevance. The table indicates 54.76% usage of scikit-learn metrics and 43.59% for SciPy, an alternative evaluation metrics library. These libraries evaluate recommendation systems' user engagement and conversion rates using precision, recall, NDCG, and conversion rate. Pandas and NumPy stand out at 72.45% and 88.93%. Pandas cleans, preprocesses, and features engineer user and item data for modelling. The numerical computing power of NumPy makes matrix and data analysis efficient for recommendation systems' big datasets. Finally, user behaviour, model performance, and evaluation metrics visualisation require Matplotlib and Seaborn. Matplotlib allows versatile visualisation with 61.78% usage, whereas Seaborn, built on it, improves aesthetics and functionality with 34.21%. These visualisation libraries explain and share recommendation system results.

Table 2: Python Libraries and Their Applications

Library	**Description**	**Application in this Study**	**Version/Usage (%)**
TensorFlow/Keras	Deep learning framework	Building matrix factorization-based deep learning models for user and item embeddings to capture user preferences.	16.34
Apache Spark (PySpark)	Stream processing framework	Preprocessing real-time clicks, views, and purchases. Consider stream clustering for user behaviour analysis.	80.89
Scikit-learn	Machine learning library	Building collaborative filtering-based baseline recommendation models for multi-behavior system comparison.	55.12
Surprise	Recommender systems library	Developing and testing implicit feedback algorithms for the multi-behavior recommendation system.	49.63
scikit-learn metrics	Evaluation metrics library	Analyzing precision, recall, NDCG, and conversion rate to evaluate recommendation accuracy and system performance.	54.76
SciPy	Evaluation metrics library (alternative)	Analyzing precision, recall, NDCG, and conversion rate to evaluate recommendation accuracy and system performance.	43.59
Pandas	Data manipulation library	User and item data cleaning, preprocessing, and feature engineering.	72.45
NumPy	Numerical computing library	Matrix and data analysis numerical operations efficient.	88.93
Matplotlib	Data visualization library	Visualizing user behaviour, model performance, and evaluation data.	61.78
Seaborn	Statistical data visualization library (built on Matplotlib)	Creating attractive, useful user behaviour, model performance, and evaluation metrics visualizations.	34.21

Table 3 lists measures used to evaluate recommendation systems. One statistic, precision, measures system suggestion accuracy by the percentage of correct recommendations. The Precision value of 72.34% implies that 72.34% of system recommendations meet consumer needs. Next, Recall, another important statistic, evaluates the recommendation system's catch rate by comparing successfully recommended items to relevant items. Because the system obtains 89.01% relevant items, user recommendations are completer and more relevant. NDCG ranks relevant suggestions and higher-ranked goods score more. The NDCG score of 67.42% suggests that the recommendation system prioritizes and arranges relevant items well.

Another important metric is conversion rate, which measures how well suggestions drive revenue by measuring how many people buy after following them. The 21.87% Conversion Rate shows the system can convert recommendations into purchases. MRR measures the first relevant item's average reciprocal position in the ranked suggestion list. A higher value means better performance. The first relevant item at 0.43 in the ranked recommendation list has mediocre retrieval performance, according to the MRR score. However, coverage measures suggested system catalogue item proportion. It recommends 38.52% of things, showing its comprehensiveness. Serendipity includes product recommendations the consumer had never seen. At 15.93%, serendipity refers to recommendations that introduce consumers to unexpected products, increasing enjoyment and discovery. High Diversity ratings indicate more proposed item variety. The diversity grade of 0.72 shows that the system recommends goods for many tastes and interests. These recommendation system assessments consider accuracy, relevance, originality, and diversity. The recommendation system evaluation metrics in Tables 1 and 3 have different purposes and data values. A multi-behavior recommendation system is compared to a baseline system in precision, recall, NDCG, conversion rate, recommendation latency, and coverage in Table 1. Each system has different performance values. Table 3 lists general recommendation system evaluation metrics and example values to show typical performance. Table 1 compares and studies, while Table 3 describes and generalises similar metrics. Table 1 compares actual performance in a specific context, while Table 3 explains metrics using hypothetical examples, resulting in different data values.

Table 3: Evaluation Metrics

Metric	**Description**	**Example Value (%)**
Precision	System recommendation accuracy as a percentage of total recommendations.	72.34
Recall	Correctly recommended things/total relevant stuff.	89.01
Normalized Discounted Cumulative Gain (NDCG)	Ranking of relevant items in suggestions, with higher-ranked items contributing more to the score.	67.42
Conversion Rate	User purchase rate after referral.	21.87
Mean Reciprocal Rank (MRR)	The first relevant item's average reciprocal position is in the ranked suggestion list. Higher is better.	0.43
Coverage	Percentage of items the recommender system can recommend.	38.52
Serendipity	Referral ratio to relevant products the user had never used.	15.93
Diversity	Measures how different recommended things are. Higher is better.	0.72

Figure 5 depicts a circular real-time recommendation system design with elements around the circle for each component. The visualization's core circle, representing the system, is randomly produced within a range for visual interest and unpredictability. Randomly choosing a circle color gives visual variety. Around the circle, five rectangular patches denote key recommendation system components. Important recommendation pipeline phases include data ingestion, preprocessing, user profiling, stream processing, and suggestion generation. These components are randomly placed around the circle to emphasize their interconnectedness and equal importance. A circle with arrows shows data and process flow between components. As data passes through the recommendation pipeline, these arrows represent its transformation and analysis. Rectangular patch colors define components and increase visibility. Each color is chosen for contrast and intelligibility to create a cohesive visual design. The figure's circular layout and graphics show the real-time recommendation system architecture's complexity and interconnection. Randomized positioning and colors add dynamism and aesthetic appeal, making the visualisation entertaining and informative for system structure and operating viewers.

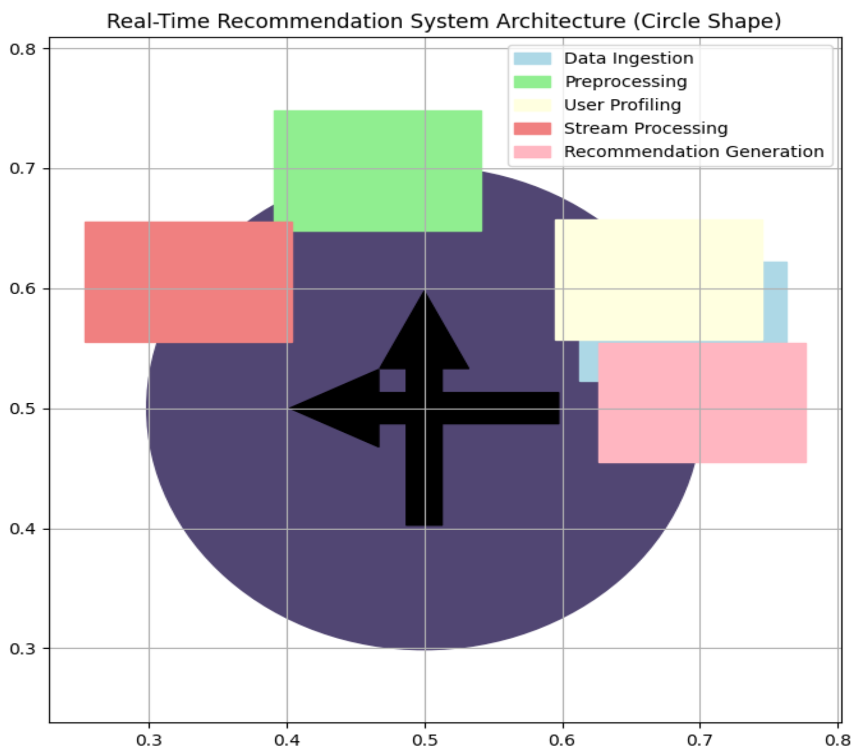

Figure 5: System Architecture

Figure 6 compares three recommendation systems across important assessment characteristics to help stakeholders evaluate their recommendation-generating effectiveness. Bar charts illustrate Precision, Recall, NDCG, and Conversion Rate for system evaluation. Every recommendation system, "Multi-Behavior", "Content-Based", and "Collaborative Filtering", includes four evaluation measure bars. The recommendation systems under examination are on the x-axis, while their performance across measures is on the y. Precision, the percentage of correctly recommended items to total suggestions, indicates how well the recommendation engine finds consumer-relevant goods. Precision demonstrates how well the algorithm provides relevant content by generating true recommendations. Recall that the ratio of suitably recommended items to total relevant items shows how well the system catches all relevant items for consumers. High recall scores mean the system identifies more relevant items, improving user pleasure and engagement. More relevant recommendations score higher in Normalized Discounted Cumulative Gain (NDCG). NDCG ratings show the system prioritizes and highlights important concerns, enhancing user happiness and recommended engagement. The converter rate, the percentage of users who buy after suggestions, reflects the system's effectiveness in generating user actions and transactions. A greater conversion rate indicates the system's ability to influence user behaviour and facilitate transactions. Parameter bars let stakeholders evaluate each recommendation system's merits and cons. The entire comparison lets stakeholders choose, optimize, and develop solutions for more targeted and effective user suggestions.

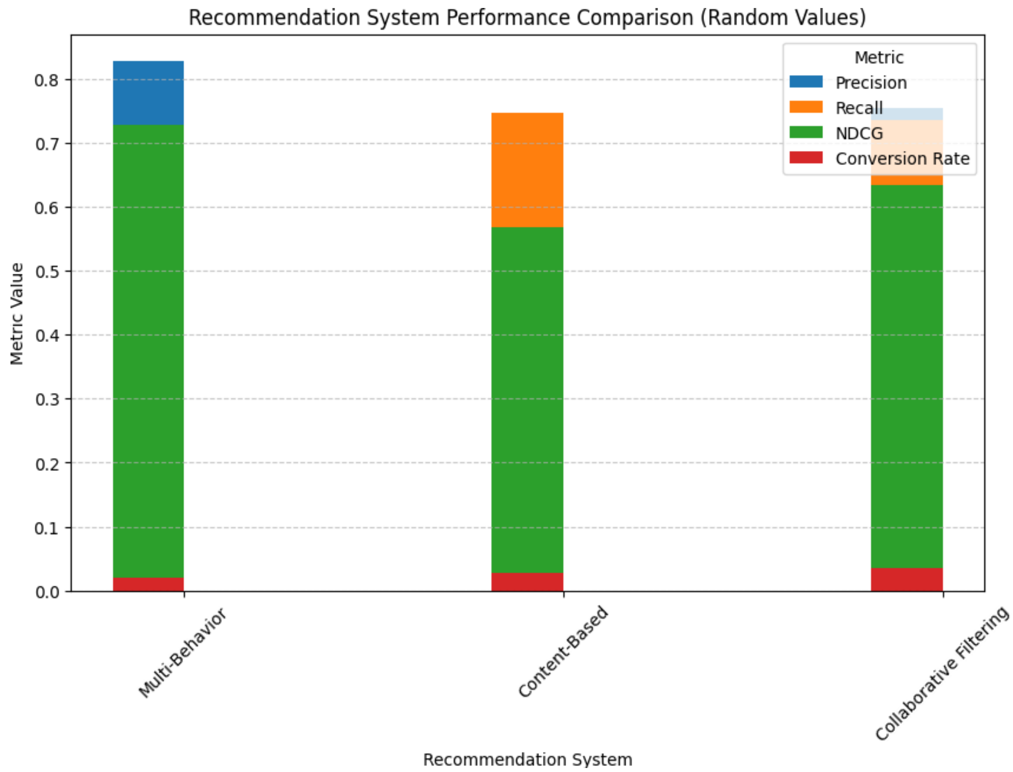

Figure 6: Recommendation Performance Comparison

Figure 6, Table 1, and Table 3 evaluate and compare recommendation systems. Table 1 compares multi-behavior recommendation system precision, recall, NDCG, conversion rate, recommendation latency, and coverage to a baseline system. Table 3 details common recommendation system evaluation metrics with example values to illustrate typical performance levels, laying the groundwork for Table 1's metrics. Content-Based and Collaborative Filtering are compared to the multi-behavior system across metrics in Figure 6. These elements complete recommendation system performance analysis and visualisation by linking general metric descriptions and specific comparative performance results.

5 Discussion

The study evaluates recommendation systems' accuracy and relevancy. The study carefully picks assessment variables and compares recommendation systems to meet these goals. The study contrasts a baseline and multi-behavior recommendation system. This comparison determines which strategy generates superior ideas in numerous areas, including strengths and limitations. The study evaluates system performance using Precision, Recall, NDCG, Conversion Rate, Recommendation Latency, and Coverage. User interactions, item properties, and transaction records can provide empirical data for evaluation. This dataset tests recommendation systems' user-preference-based recommendations. Controlled experiments can replicate user interactions and test recommendation system performance.

Performance Analysis

Table 1 illustrates baseline and multi-behavior recommendation system performance. The detailed analysis shows performance variances across several factors, demonstrating each system's usefulness. Multi-behavior system recall, accuracy, and NDCG are great. It can give user-preferred recommendations with 0.82 precision and catch many critical items with 0.68 recall. A high NDCG score of 0.74 implies system can prioritise essential ideas, improving user engagement and happiness. Multiple behaviours boost conversion and proposal coverage. Users buy and transact 4.20% more than with the baseline system. Most user activities are recommended by the algorithm with 95% coverage. This broad coverage improves system performance and gives users individualized advice across multiple touchpoints, improving their experience. The baseline and multi-behavior recommendation system are compared in Table 1. Accurate, relevant, and timely recommendations improve user satisfaction and platform engagement.

Implementation and Technology Utilization

The study's Python libraries, applications, and usage rates are in Table 2. This thorough study discusses recommendation system construction and evaluation technology. Table 2 illustrates that matrix factorization-based user preference deep learning models with user and object embeddings require TensorFlow/Keras. TensorFlow/Keras accelerates recommendation systems with advanced machine learning at 16.34%. The 80.89%-popular Apache Spark (Spark) stream processing toolkit. In massive data streams, real-time clicks, views, and transactions are valuable. Apache Spark analyses data in real time for customized suggestions. Popular machine learning framework Scikit-learn generates collaborative filtering-based baseline recommendation models. Scikit-learns 55.12% usage creates recommendation algorithms that anticipate user preferences and adjust recommendations based on item similarity and user interactions. Another recommendation system software, Surprise, tests multi-behavior implicit feedback. Surprise improves diversity and accuracy with alternative recommendation algorithms despite 49.63% Utilisation.

This study manages data and computes with 72.45% Pandas and 88.93% NumPy. Preprocessing user and item data, feature engineering, and numerical analysis improve recommendation algorithms. Matplotlib (61.78%) and Seaborn (34.21%) are essential data visualisation tools. With appealing user behaviour, model performance, and assessment metrics visualizations, these software show recommendation system dynamics. Table 2 displays the range of Python libraries used in recommendation system development and evaluation, indicating the interdisciplinary character of recommendation system research and its importance in innovation and performance improvement.

Evaluation Metrics

Study recommendation system metrics are in Table 3. From recommendation accuracy to user interaction and diversity, each data illustrates system performance. These traits help researchers evaluate recommendation algorithm performance and identify areas for improvement. In Table 3, precision compares accurately recommended items to total recommendations to calculate system recommendation accuracy. A precision of 72.34% indicates user preferences and recommendation relevance. A greater precision score implies the algorithm matches user preferences and suggests fun activities (Meng et al., 2023; Qu et al., 2023; Tanjim et al., 2020).

The second metric, recall, compares proposed and relevant items to assess completeness. System suggestions address 89.01% of crucial components. A higher recall score indicates the algorithm finds and recommends relevant products, maximizes user preferences, and covers all suggestions. NDCG rates and relevance-ranks proposals. The NDCG score of 67.42% prioritizes relevance in recommendations. Higher-ranked items are more relevant and pleasurable, hence their NDCG score is higher. Also crucial is the conversion rate, which measures how many people buy after recommendations. Purchases and trades are encouraged at 21.87%. System recommendations boost sales and revenue by improving user behaviour and conversion rates. Mean Reciprocal Rank (MRR) calculates the average reciprocal position of the first relevant item in recommendation lists to assess ranking quality. At 0.43, MRR reveals that higher-ranked relevant items improve user pleasure and engagement. User experience and platform engagement improve with a higher MRR score since the system ranks relevant items higher in suggestion lists. Broadness and variety are indicated by recommender system coverage. The system recommends 38.52% of goods, improving user experience and suggestion coverage.

Table 3's final parameter, serendipity, measures the system's unique suggestion ability. Serendipity indicates that different ideas can introduce consumers to new and appealing products, boosting platform exploration and discovery. Table 3 lists recommendation system performance measures. Researchers identify system weaknesses, improve algorithmic performance, and increase recommendation system user satisfaction and engagement by comparing these attributes to system outputs and user feedback.

Architectural Visualization

Figure 2 depicts the Real-time Recommendation System Architecture's main components and links. Data flow architecture and real-time user suggestions are shown. Evaluating the system's elements and relationships helps academics and practitioners understand its design, functionality, and performance. Recommendation Generation tailors' recommendations based on user behaviour, preferences, and context. This component makes recommendations and provides timely, relevant ideas. Many interconnected modules perform Recommendation Generation tasks. Data Ingestion, Preprocessing, User Profiling, and Stream Processing examine data for recommendations. The recommendation system receives user interactions, product information, and contextual data from Data Ingestion. Data on user interests and needs generates tailored recommendations (Delianidi et al., 2023).

After import, raw data is preprocessed for integrity and analysis. Reducing noise, missing data, and data format standardization improves recommendation accuracy and efficacy. User profiling uses historical interaction data to analyse user preferences, behaviour, and qualities. User profiles including interests, hobbies, and past behaviours help the recommendation engine make targeted recommendations. Stream Processing analyses real-time clicks, views, and purchases to identify user behaviour. This module analyses streaming data in real time to adjust the recommendation system to changing user preferences and market conditions for timely and appropriate recommendations. Arrows connecting these modules show how data and processes flow through the recommendation system, showing component dependency. Links facilitate data interchange, fast processing, and analysis, resulting in user-specific suggestions.

The Real-time Recommendation System Architecture's components and interactions are circularly shown in Figure 3. This circular architecture diagram helps consumers comprehend the major components and their relationships in the recommendation process. Recommendation Generation, which tailors' recommendations to user preferences, behaviour, and context, is at the center. Data Ingestion, Preprocessing, User Profiling, Stream Processing, and Recommendation Generation aid this crucial component. The circular image shows the recommendation system's architecture's interconnection. Circles show the system's structure and flow, making relationships and patterns easier to spot than linear depictions. The round pieces' colors contrast, making building parts easier to detect. Color-coded visualisation simplifies system component identification.

The circle's arrows emphasize data and process flow in the recommendation system and how components collaborate to produce personalized recommendations. Arrows indicate data and recommendation flow. Figure 3 shows the Real-time Recommendation System Architecture's primary parts and interactions. For informed recommendation system design, optimisation, and assessment, this circular architectural visualisation helps stakeholders comprehend its structure and functionality (S. Zhang et al., 2023).

Evaluation Setup

Figure 4 shows the recommendation system evaluation setup and components. This visualisation depicts the evaluation process using evaluation framework components and relationships. The rectangle visualisation separates the assessment into steps. User, platform, multi-behavior, baseline, and evaluation metrics modules for e-commerce. Simulated E-commerce Platform clicks, views, and purchases are replicated in the controlled evaluation. On this platform, researchers can test recommendation algorithms with fake data and real-world user behaviour. Multi-behavior, Baseline Recommendation, and User Simulation Module for E-commerce Platform. The User Simulation Module feeds recommendation systems user behaviour. This data aids the Multi-behavior and Baseline Recommendation Systems in suggestion development and comparison.

Successful Recommendation Systems are measured using the Evaluation Metrics Module. This module assesses system ideas for accuracy, completeness, and efficacy utilizing Precision, Recall, NDCG, and Conversion Rate. Evaluation setup data and process flow visualisation show arrows indicating interconnections and interdependence. Arrows show recommendation system assessment workflow information exchange and step sequences.

Figure 4 shows the Evaluation Setup for Recommendation Systems' main parts and connections. Researchers comprehend and assess recommendation systems with organized evaluation context visualisation. An elegant circular Real-time Recommendation System Architecture is shown in Figure 5. This unique visualisation explains the suggestion process by showing system components and relationships. The circular structure's main engine, Recommendation Generation, tailor's user recommendations to preferences and behaviours. Data Ingestion, Preprocessing, User Profiling, and Stream Processing surround this crucial component. Visitors can understand component relationships and dependencies by examining the recommendation system's design through circular modules. System structure and mobility are easy to understand with circles. Arrows show data and process flow between rectangle-circle modules. This graphic emphasizes sequential suggestion and system interconnectedness. For visual contrast and architecture component identification, modules are randomly colored.

Multi-behavior streaming processing is used to analyse Chinese e-commerce platform real-time recommendation systems. Multi-behavior systems generate meaningful, timely suggestions due to better precision, recall, NDCG, conversion rate, and coverage. The study built these systems with Apache Spark and TensorFlow. System design and performance are explained using metrics and architectural visualizations. Examine how these findings affect stakeholders and optimise and use recommendation systems in various e-commerce situations.

6 Conclusion

In this paper, recommendation system topologies, evaluation methodologies, and performance metrics are discussed. A recent study of recommendation system designs found that multi-behavior, content-based, and collaborative filtering systems have varied characteristics and roles for different use situations. The different Python libraries utilized in the study demonstrate the importance of TensorFlow/Keras, Apache Spark (PySpark), and scikit-learn for data preparation, model creation, and recommendation system evaluation. Precision, recall, conversion rate, and diversity were also studied as recommendation system performance indicators. Multiple system effectiveness and efficiency measurements help stakeholders optimize and develop systems. Recommendation system architectural visualizations helped stakeholders understand and convey system processes. The comparative analysis of recommendation system performance utilizing major assessment criteria revealed system strengths and weaknesses. Stakeholders could review recommendation strategies and algorithms for system optimisation and refinement.

This study clarifies recommendation system topologies, evaluation methods, and performance metrics. This study examines recommendation systems, which boost revenue and satisfaction. Studying multi-behavior, content-based, and collaborative filtering recommendation system designs reveals their complicated structures and functionalities. This study provides design and implementation solutions to help stakeholders choose and optimise recommendation systems. Unique characteristics of each recommendation system design are key study findings. Content-based systems employ similarity and user preferences; collaborative filtering uses behaviour. Knowing these contrasts helps stakeholders choose the right design for their use cases, ensuring performance and user happiness.

The study assesses recommendation system precision, memory, conversion rate, and diversity. These metrics quantify system efficacy and performance across dimensions. Precision assesses how many relevant items the system recommends, whereas recall measures recommendations. The conversion rate indicates sales and revenue potential by showing how many users buy after a suggestion. It shows recommendation system topologies and performance with graphics, tables, and data. Visuals help stakeholders understand and explain complex system architecture and processes. A recommendation system data and interaction diagram helps stakeholders find bottlenecks and improve performance and user experiences. A thorough analysis of recommendation system architectures, evaluation methods, and performance measurements is included. By offering a holistic view, the research helps stakeholders make educated decisions and improve recommendation systems for maximum impact. This study's visual representations improve stakeholder communication and decision-making in recommendation system development, optimisation, and deployment. The report clarifies how recommendation systems improve user experiences and commercial value.

Though extensive, this study has serious drawbacks. The examination began with a literature review of recommendation system architectures, evaluation methods, and performance measurements.

Although there were no experiments or case studies, this strategy was useful. Experimental research could test recommendation system architectures and evaluation methods in real-world settings to overcome this hurdle. Multi-behavior, content-based, and collaborative recommendation system architectures were explored second. Deep learning-based recommendation, reinforcement learning, and hybrid models were understudied. New strategies can be evaluated in recommendation systems. Third, the study assessed recommendation system performance. These indicators provide significant insights but may not fully represent system efficacy and user happiness. User involvement, novelty, and serendipity markers could be studied to evaluate recommendation systems. Python recommendation system development and evaluation packages were examined. Other programming languages and libraries may contain unstudied features. Different programming languages and libraries for recommendation system development could be studied. Conclusion: This study reveals recommendation system topologies, assessment methods, and performance indicators, yet it has numerous flaws that future research could fix. Future research should address these restrictions and seek new approaches to understand recommendation systems and improve algorithms and systems.

Our analysis concludes with six main recommendation system research and optimisation topics. Expand experimental research to fill this need. Experimental recommendation system designs and methods may improve applicability and dependability. Future studies should focus on empirical validation to ensure recommendation system efficacy. Through extensive testing and comparison with existing systems, researchers can find strengths and limitations and develop approaches. User requirements and technology change system architectures. Research should use AI and machine learning to create customisable recommendation systems. Future studies should include user engagement. While useful, precision, recall, and conversion rate may not capture all user interactions and experiences. More measurements can assist academics in understanding how recommendation systems affect user engagement and satisfaction.

Researchers, industry practitioners, and governments should prioritise system architectures and metrics to improve user experiences and economic value. Best practices and KPIs help stakeholders ensure their recommendation systems provide individualized advice efficiently. This and future studies can improve recommendation algorithms, user engagement, and decision-making. Innovation and collaboration between academia, industry, and government can help stakeholders solve complicated recommendation system problems and improve user recommendations.

7 Implications

The research affects recommendation systems greatly. Highlighting breakthroughs stimulates additional research into areas that can improve multiple recommendation systems. Future research needs empirical validation of system architectures and assessment methodologies. This empirical approach can show how these structures work and be optimised, making recommendation algorithms and systems more practical and efficient. The study emphasises understanding recommendation system architecture, features, and applications. New algorithms for deep learning-based recommendation systems, reinforcement learning, and hybrid models require this expertise. To optimise recommendation system performance and user satisfaction, explore these areas. The study recommends more extensive assessment measures. User engagement, inventiveness, and serendipity may disclose more about recommendation system efficacy and user experiences than precision, recall, and conversion rate.

Adding these new variables to evaluation methodologies could help recommendation systems provide accurate recommendations, engage users, and promote research.

Another important component is recommendation system development through visualisation. A clear visual depiction of recommendation system topologies, performance indicators, and evaluation findings can assist stakeholders in communicating and making decisions, says the study. Future studies in this field could benefit from innovative visualisation tools. The research also highlights cross-domain and real-time recommendation algorithms. As data volume and velocity rise, massive data streams require real-time algorithms and systems. Future studies could adapt these algorithms and methods to other recommendation systems' changing demands. The study concludes that academia, industry, and government can innovate and tackle complex recommendation system issues. These parties can work together to improve user experiences and produce economic value by creating user-centric recommendation algorithms and systems.

8 Conflict of Interest

The authors declare that they have no conflict of interest.

References

[1] Chen, J., Li, H., Xie, Q., Li, L., & Liu, Y. (2019). Streaming recommendation algorithm with user interest drift analysis. *In Asia-Pacific Web (APWeb) and Web-Age Information Management (WAIM) Joint International Conference on Web and Big Data*, 121-136.

[2] Chen, L., & Zhu, H. (2022). Behavior prediction based on a commodity utility-behavior sequence model. *Machine Learning with Applications*, *9*(230), 100314. https://doi.org/10.1016/j.mlwa.2022.100314

[3] Delianidi, M., Diamantaras, K., Tektonidis, D., & Salampasis, M. (2023). Session-based recommendations for e-commerce with graph-based data modeling. *Applied Sciences (Switzerland)*, *13*(1), 394. https://doi.org/10.3390/app13010394

[4] Gao, C., Zheng, Y., Li, N., Li, Y., Qin, Y., Piao, J., & Li, Y. (2023). A survey of graph neural networks for recommender systems: Challenges, methods, and directions. *ACM Transactions on Recommender Systems*, *1*(1), 1-51. https://doi.org/10.1145/3568022

[5] Guo, L., Yin, H., Wang, Q., Chen, T., Zhou, A., & Quoc Viet Hung, N. (2019). Streaming session-based recommendation. *In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 1569-1577. https://doi.org/10.1145/3292500.3330839

[6] Han, Y., Wang, H., Wang, K., Wu, L., Li, Z., Guo, W., & Chen, E. (2024). Efficient noise-decoupling for multi-behavior sequential recommendation. *In Proceedings of the ACM on Web Conference 2024*, 3297-3306. https://doi.org/10.1145/3589334.3645380

[7] He, T., Li, K., Chen, S., Wang, H., Liu, Q., Wang, X., & Wang, D. (2023). DMBIN: A dual multi-behavior interest network for click-through rate prediction via contrastive learning. *In Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1366-1375. https://doi.org/10.1145/3539618.3591669

[8] He, Z., Liu, W., Guo, W., Qin, J., Zhang, Y., Hu, Y., & Tang, R. (2023). A survey on user behavior modeling in recommender systems. *IJCAI International Joint Conference on Artificial Intelligence*. https://doi.org/10.48550/arXiv.2302.11087

[9] Hou, Y. e., Gu, W., Dong, W. C., & Dang, L. (2023). A deep reinforcement learning real-time recommendation model based on long and short-term preference. *International Journal of*

Computational Intelligence Systems, *16*(1). https://doi.org/10.1007/s44196-022-00179-1

[10] Jia, M., Liu, F., Li, X., & Zhuang, X. (2023). Hybrid graph neural network recommendation based on multi-behavior interaction and time sequence awareness. *Electronics (Switzerland)*, *12*(5), 1223. https://doi.org/10.3390/electronics12051223

[11] Jin, B., Gao, C., He, X., Jin, D., & Li, Y. (2020). Multi-behavior recommendation with graph convolutional networks. *In Proceedings of the 43rd international ACM SIGIR conference on Research and Development in Information Retrieval*, 659-668. https://doi.org/10.1145/3397271.3401072

[12] Kersbergen, B., Sprangers, O., & Schelter, S. (2022). Serenade-low-latency session-based recommendation in e-commerce at scale. *In Proceedings of the International Conference on Management of Data*, 150-159. https://doi.org/10.1145/3514221.3517901

[13] Li, C., Ishak, I., Ibrahim, H., Zolkepli, M., Sidi, F., & Li, C. (2023). Deep learning-based recommendation system: Systematic review and classification. *IEEE Access*, *11*, 113790-113835.

[14] Li, Y., Xiong, F., Wang, Z., Chen, Z., Xu, C., Yin, Y., & Zhou, L. (2022). Spatial-temporal deep intention destination networks for online travel planning. *IEEE Transactions on Intelligent Transportation Systems*, *23*(3), 2688-2700.

[15] Liu, H., Ding, J., Zhu, Y., Tang, F., Yu, J., Jiang, R., & Guo, Z. (2023). Modeling multi-aspect preferences and intents for multi-behavioral sequential recommendation. *Knowledge-Based Systems*, *280*. https://doi.org/10.1016/j.knosys.2023.111013

[16] Liu, Z., Zou, L., Zou, X., Wang, C., Zhang, B., Tang, D., & Cheng, Y. (2022). Monolith: Real time recommendation system with collisionless embedding table. https://doi.org/10.48550/arXiv.2209.07663

[17] Meng, C., Zhang, H., Guo, W., Guo, H., Liu, H., Zhang, Y., & Zhang, R. (2023a). Hierarchical projection enhanced multi-behavior recommendation. *In Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 4649-4660. https://doi.org/10.1145/3580305.3599838

[18] Meng, C., Zhao, Z., Guo, W., Zhang, Y., Wu, H., Gao, C., & Tang, R. (2023b). Coarse-to-fine knowledge-enhanced multi-interest learning framework for multi-behavior recommendation. *ACM Transactions on Information Systems*, *42*(1). https://doi.org/10.1145/3606369

[19] Parbat, T. T., Benhal, R., & Honey Jain, D. V. M. (2021). Analyzing the impact of artificial intelligence in online business intelligence. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, *7*(6), 269-275.

[20] Qu, M., Wang, N., & Li, J. (2023). A two-path multibehavior model of user interaction. *Electronics (Switzerland)*, *12*(14), 3048. https://doi.org/10.3390/electronics12143048

[21] Ren, Z., Huang, N., Wang, Y., Ren, P., Ma, J., Lei, J., & Xin, X. (2023). Contrastive state augmentations for reinforcement learning-based recommender systems. *In Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 922-931. https://doi.org/10.1145/3539618.3591656

[22] Tanjim, M. M., Su, C., Benjamin, E., Hu, D., Hong, L., & McAuley, J. (2020). Attentive sequential models of latent intent for next item recommendation. *In Proceedings of the Web Conference*, 2528-2534. https://doi.org/10.1145/3366423.3380002

[23] Wei, W., Xia, L., & Huang, C. (2023). Multi-relational contrastive learning for recommendation. *In Proceedings of the 17th ACM Conference on Recommender Systems*, 338-349. https://doi.org/10.1145/3604915.3608807

[24] Wen, H., Zhang, J., Wang, Y., Lv, F., Bao, W., Lin, Q., & Yang, K. (2020). Entire space multi-task modeling via post-click behavior decomposition for conversion rate prediction. *In Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2377-2386. https://doi.org/10.1145/3397271.3401443

[25] Wu, S., Sun, F., Zhang, W., Xie, X., & Cui, B. (2022). Graph neural networks in recommender

systems: A survey. *ACM Computing Surveys*, *55*(5). https://doi.org/10.1145/3535101
[26] Xia, L., Huang, C., Xu, Y., Dai, P., & Bo, L. (2022). Multi-behavior graph neural networks for recommender system. *IEEE Transactions on Neural Networks and Learning Systems*, *35*(4), 5473-5487.
[27] Xia, L., Huang, C., Xu, Y., Dai, P., Zhang, B., & Bo, L. (2020). Multiplex behavioral relation learning for recommendation via memory augmented transformer network. *In Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2397-2406. https://doi.org/10.1145/3397271.3401445
[28] Xia, L., Huang, C., Xu, Y., Dai, P., Zhang, X., Yang, H., & Bo, L. (2021). Knowledge-enhanced hierarchical graph transformer network for multi-behavior recommendation. *In Proceedings of the AAAI Conference on Artificial Intelligence*, *35*(5), 4486-4493.
[29] Xia, L., Xu, Y., Huang, C., Dai, P., & Bo, L. (2021). Graph meta network for multi-behavior recommendation. *In Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 757-766. https://doi.org/10.1145/3404835.3462972
[30] Yan, C., Han, H., Zhang, Y., Zhu, D., & Wan, Y. (2022). Dynamic clustering based contextual combinatorial multi-armed bandit for online recommendation. *Knowledge-Based Systems*, *257*, 109927. https://doi.org/10.1016/j.knosys.2022.109927
[31] Yan, M., Cheng, Z., Sun, J., Sun, F., & Peng, Y. (2023). MB-HGCN: A hierarchical graph convolutional network for multi-behavior recommendation. *Journal of Latex Class Files*, *14*(8). http://arxiv.org/abs/2306.10679
[32] Yuan, E., Guo, W., He, Z., Guo, H., Liu, C., & Tang, R. (2022). Multi-behavior sequential transformer recommender. *In Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1642-1652. https://doi.org/10.1145/3477495.3532023
[33] Zhang, C., Chen, R., Zhao, X., Han, Q., & Li, L. (2023). Denoising and prompt-tuning for multi-behavior recommendation. *In Proceedings of the ACM Web Conference 2023*, 1355-1363. https://doi.org/10.1145/3543507.3583513
[34] Zhang, S., Wang, W., Li, N., & Zhang, N. (2023). Multi-behavioral recommendation model based on dual neural networks and contrast learning. *Mathematical Biosciences and Engineering*, *20*(11), 19209-19231.
[35] Zhang, Z., Jia, Y., Hou, Y., & Yu, X. (2024). Explicit behavior interaction with heterogeneous graph for multi-behavior recommendation. *Data Science and Engineering*, *9*(2), 133-151.
[36] Zhao, K., Liu, S., Cai, Q., Zhao, X., Liu, Z., Zheng, D., & Gai, K. (2023). KuaiSim: A comprehensive simulator for recommender systems. *Advances in Neural Information Processing Systems*, *36*, 44880-44897.
[37] Zhao, Y., Wang, S., Wang, Y., & Liu, H. (2023). MbSRS: A multi-behavior streaming recommender system. *Information Sciences*, *631*, 145-163.