International Journal of Computational and Experimental Science and ENgineering (IJCESEN)

Vol. 11-No.3 (2025) pp. 4315-4328
http://www.ijcesen.com

ISSN: 2149-9144

Research Article

PRE-ADDL: Optimized Attention-Driven Deep Learning Mechanisms for Accurate and Computationally Efficient E-Commerce Recommendations

Supriya Saxena[1, 2*], Bharat Bhushan[3]

[1]Research Scholar, Department of Computer Science and Engineering, Sharda University, Greater Noida, India and
[2]Assistant Professor, IT Department, Hierank Business School, Noida, India
*** Corresponding Author Email:** supriya08.saxena@gmail.com - **ORCID**: 0009-0005-5967-2447

[3]Associate Professor, Department of Computer Science and Engineering, Sharda University, Greater Noida, India
Email: bharat_bhushan1989@yahoo.com - **ORCID:** 0000-0002-9345-4786

Article Info:

DOI: 10.22399/ijcesen.2846
Received : 15 March 2025
Accepted : 12 June 2025

Keywords

E-commerce
Recommended system
Deep Learning
Attention
Optimizer
Web Usage mining

Abstract:

In directing users' decision-making across a variety of online platforms, recommendation systems are essential. Enhancing the accuracy and relevance of these systems has become a more significant challenge in both academic research and industry applications as the volume of web data keeps growing. Although many models have been developed to address this issue, the effectiveness of many traditional approaches can be hampered by their reliance on narrow perspectives. Using web usage mining techniques, we present an Attention-Driven Deep Learning Model based recommendation system in this study. In order to provide more precise and tailored recommendations, our method looks for intricate patterns and connections in user behaviour and online interactions. We evaluated our approach on public web log datasets, using a temporal evaluation protocol that simulates the dynamics of an E-commerce website in a realistic way. The study found that although more than 1.4 million users engaged with products, just 0.83% of them became buyers, which indicates the difficulty of enhancing engagement and conversion rates. A deep Learning model utilizing an attention mechanism was built to improve personal recommendations. The architecture of the model involves various layers, i.e., embedding, attention, feature extraction, and dense layers, to effectively capture user-item interactions. Experimental results showed that the model reported approx. 97% accuracy with excellent Precision and Recall. The recommendation system efficiently yielded top 5 product recommendations to users, where relevant items recorded probability scores ranging up to 0.0257. Computational efficiency is revealed through the 0.63 seconds response time. The study's findings highlight how well deep learning can enhance user engagement and streamline personalized suggestions, creating opportunities for additional advancements in e-commerce recommendation systems.

1. Introduction

In today's era of rapidly expanding internet access and the prevalence of smart phones, we are constantly connected to social media, e-commerce platforms, and a myriad of online information sources. This continuous generation and dissemination of content result in vast amounts of web data, much of which remains unexplored and unrefined. The overwhelming volume of information can lead to confusion and chaos for users. To navigate this complexity, e-commerce giants are leveraging this data for analytics, developing more effective recommendation systems to enhance user experience [1]. The traffic generated from online activities, such as posting and downloading, produces substantial log data. This data can be harnessed through web mining techniques for web analytics and the development of recommendation systems. Web mining can be categorized into three main types: web usage mining, web content mining, and web structure mining. As more individuals browse the internet and engage in e-commerce transactions, this dataset is continuously updated, making it a prime example of big data. One of the primary applications of web data is in the creation of recommendation systems (RS) [2]. This specialized field of machine learning focuses on making predictions based on user preferences. Recommendation systems are vital for the sales performance of any e-commerce platform, as users tend to favor websites that understand their needs and suggest relevant products [3]. The recommendations provided by these systems are influenced by various factors, including user demographics, search history, purchase history, and browsing patterns [4].

Web usage mining is a part of data mining that deals specifically with examining user behavior with websites in order to reveal useful patterns. Web usage logs are generated and recorded with every click made by users. These logs are stored in three key locations: at the user level as browser logs, at the proxy level as proxy logs, and finally at the server level as server logs. Analyzing these web logs is essential for understanding user behavioral patterns, such as navigation paths through a website, the most frequently visited pages, and peak traffic load times. This analysis provides crucial insights for web design optimization, the development of user recommendation systems for e-commerce, targeted advertising, and relevant product recommendations. Additionally, it aids in identifying trends within user groups [5]. Through data gathering and processing from web logs, cookies, and browsing history, it assists in comprehending user behavior, forecasting trends, and enhancing website functionality. Data collection, preprocessing, pattern discovery, and analysis are all the steps involved, which eventually benefit businesses by optimizing their web presence. Web usage mining is comprehensively applied to personalization, targeted advertising, and fraud analysis, thus ranking as a cornerstone tool in online environments. Figure 1 gives a glimpse of Web usage mining.

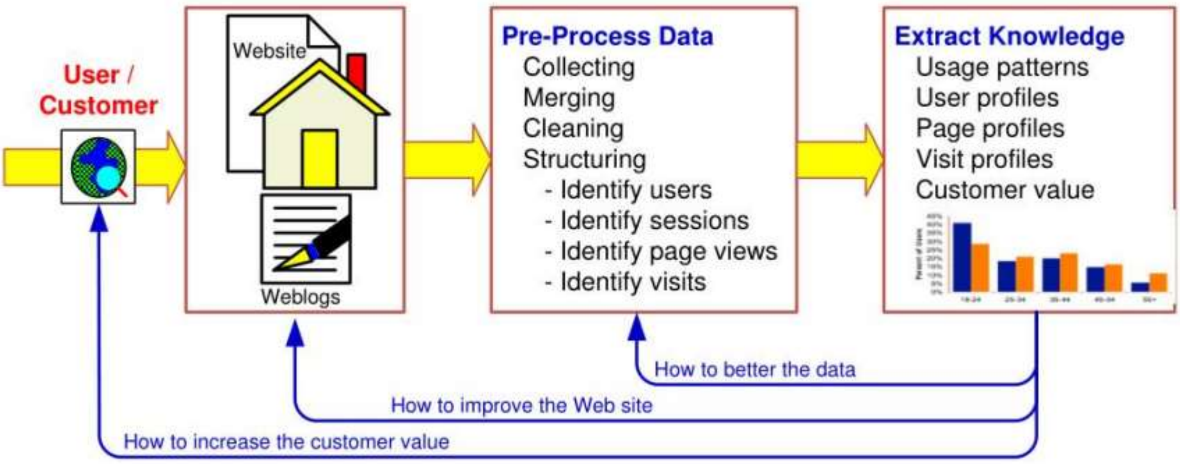

Figure 1. *Overview of Web usage mining*

Recommendation systems are smart programs designed to suggest related content, services, or items based on an individual's tendencies and preferences [6]. Such systems utilize various techniques, including content-based filtering, collaborative filtering, and hybrid approaches, to maximize user interactivity and satisfaction. Online marketplaces like Amazon, video streaming platforms such as Netflix, and music streaming services like Spotify have integrated recommender systems as critical components that enhance user experience [7]. By combining web usage mining with recommendation systems, businesses can analyze user behavior and preferences more effectively, enabling them to provide more relevant and personalized suggestions. This integration allows for a deeper understanding of user interactions, leading to improved accuracy in recommendations and ultimately fostering greater customer loyalty and engagement.

In machine learning, recommendation systems fall into two primary categories: content-based filtering and collaborative filtering [9]. Content-based recommendation systems look at the characteristics and attributes of the items to create suggestions. For example, based on how similar their content is, the algorithm would recommend additional action movies to a user who has indicated interest in them [9]. Conversely, recommendation systems that employ collaborative filtering place more emphasis on the behaviors and interests of similar users. By identifying trends among users with similar interests, collaborative filtering can suggest things that individuals with similar preferences have enjoyed [10].

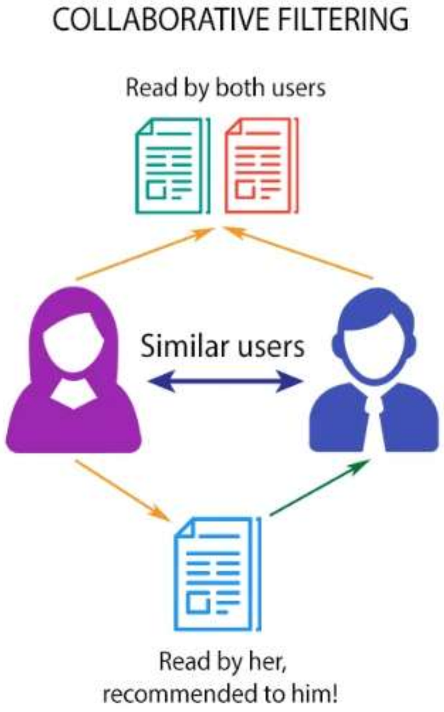

Figure 2. *Overview of Recommendation systems*

The rest of the paper is structured as follows. Section 2 investigates the related work. Section 3 introduces the proposed model. Section 4 presents the simulation results and discussion. Section 5 concludes the paper.

2. Literature Review

Sameena et al. (2025) [11] developed a personalized product recommendation model for e-commerce utilizing the H&M dataset. The study explored various recommendation techniques, including KNN Basic, Non-negative Matrix Factorization (NMF), Co-Clustering, and Singular Value Decomposition (SVD). By optimizing hyperparameters, the research found that the SVD model achieved the highest accuracy, reaching 90.40% with a low Root Mean Square Error (RMSE) of 0.2261. The study's findings were limited to the H&M dataset, raising concerns about its applicability to other e-commerce platforms.
Trinh et al. (2025) [12] aimed to develop a scalable collaborative filtering recommender system using Apache Spark. By leveraging distributed computing, the system achieved a 7.6x speedup in training while maintaining an RMSE of 0.9. The study demonstrated significant improvements in efficiency, making large-scale recommendation models more practical for e-commerce applications. Despite these advancements, further enhancements in accuracy were suggested through the integration of multi-model approaches.
Vashishth et al. (2025) [13] focused on analyzing the impact of AI-driven content personalization on customer experience in e-commerce. By conducting a literature review and case study analysis, the research highlighted how AI enhances customer satisfaction, engagement, and conversion rates. The study underscored the benefits of AI-powered personalization but also pointed out challenges such as data privacy concerns and the need for continuous algorithm refinement to maintain accuracy and relevance.
Wu et al. (2025) [14] introduced an Interest Unit (IU)-based product recommendation approach tailored for consumer-to-consumer (C2C) platforms. The methodology involved clustering products based on shared attributes and implementing a two-stage IU-based recommendation framework. Results indicated that persistent IU behaviors significantly improved recommendation quality compared to item-specific interactions. However, the approach might not be suitable for platforms where item-based purchase patterns dominate.
Krishna et al. (2025) [15] sought to improve e-commerce recommendations by integrating sentiment analysis with collaborative filtering. The study introduced the Multi-Layer Attention-based Encoder-Decoder Temporal CNN (MLA-EDTCNet) model, enhancing recommendation accuracy through sentiment polarity analysis. By incorporating sentiment-aware filtering, the model outperformed state-of-the-art techniques. Nevertheless, class imbalance issues were identified, which were addressed using a Modified Conditional Generative Adversarial Network (MCGAN), though further real-world testing was recommended.
Bahi et al. (2025) [16] proposed enhancing recommendation diversity through the use of Siamese networks and clustering. Their model employed a Siamese network alongside ResNet for feature extraction, followed by clustering to increase diversity in recommendations. The approach achieved high accuracy (88.5%), precision (90.2%), and recall (87.1%) while maintaining relevance. However, the computational complexity of training and implementation posed a significant limitation, requiring substantial resources.
Huang et al. (2025) [17] optimized personalized product recommendations by considering stochastic purchase probability. The study introduced a two-stage recommendation system, incorporating logistic regression to model user preferences and stochastic optimization to maximize revenue. The approach balanced recommendation accuracy with revenue generation, making it advantageous for e-

commerce platforms seeking financial optimization. However, the stochastic modeling techniques used may not fully capture the complexities of real-world purchasing behavior.

Sharma and Paço (2025) [18] explored factors influencing green purchasing behavior in e-commerce through the O ZONE model and SOBC framework. The study employed Partial Least Squares Structural Equation Modeling (PLS-SEM) and SmartPLS analysis to assess self-efficacy and awareness regarding green products. Findings provided insights into consumer behaviour toward sustainable purchases. However, the reliance on self-reported data introduced potential biases, impacting the study's generalizability.

Li et al. (2025) [19] advanced agricultural product recommendations using multimodal AI. Their model incorporated LLAVA for data enhancement, fused product association relationships, and extracted user modal preferences to refine suggestions. Testing on an Amazon dataset showed superior recommendation accuracy compared to baseline models. However, since the study primarily focused on agricultural products, its applicability to other e-commerce categories remained uncertain.

Ahmad (2025) [20] investigated the role of digital technologies in improving supply chain efficiency in e-commerce. Using a qualitative research approach involving interviews, focus groups, and document analysis, the study highlighted the benefits of AI, IoT, blockchain, and big data analytics in logistics and inventory management. Despite these insights, the lack of quantitative performance metrics limited the ability to assess the precise impact of these technologies on supply chain efficiency.

Table 1. *Previous work done in domain of E commerce*

Ref	Objective	Methodology	Advantage	Limitations
[11] Sameena et al. (2025)	Develop and evaluate a personalized product recommendation model for e-commerce.	Utilized H&M dataset; applied KNNBasic, NMF, Co-Clustering, and SVD; optimized hyperparameters for better accuracy.	SVD model achieved high accuracy (90.40%) and low RMSE (0.2261), demonstrating the importance of hyperparameter tuning.	Limited to specific dataset (H&M); results may not generalize to other e-commerce platforms.
[12] Trinh et al. (2025)	Develop a scalable product-based collaborative filtering recommender system.	Implemented Apache Spark for distributed computing, achieving 7.6x speedup while maintaining RMSE of 0.9.	Improved training speed significantly while maintaining recommendation accuracy.	Future work needed to enhance accuracy further with multi-model approaches.
[13] Vashishth et al. (2025)	Analyze AI-driven content personalization's impact on customer experience in e-commerce.	Literature review, case studies, and empirical evidence analysis.	Demonstrates how AI improves customer satisfaction, engagement, and conversion rates.	Challenges include data privacy concerns and the need for continuous algorithm refinement.
[14] Wu et al. (2025)	Propose Interest Unit-based product recommendation to improve recommendations on C2C platforms.	Grouped products into clusters based on attributes; introduced a two-stage IU-based recommendation framework.	More persistent IU behaviors enhance recommendation quality compared to item-specific interactions.	May not be suitable for platforms with strong item-based purchase patterns.
[15] Krishna et al. (2025)	Improve e-commerce recommendations using sentiment analysis and collaborative filtering.	Integrated Multi-Layer Attention-based Encoder-Decoder Temporal CNN (MLA-EDTCNet) with sentiment analysis and collaborative filtering.	Enhanced recommendation accuracy using sentiment polarity; outperformed state-of-the-art models.	Class imbalance issues addressed via MCGAN, but further real-world testing needed.
[16] Bahi et al. (2025)	Improve recommendation diversity using Siamese networks and clustering.	Utilized Siamese network and ResNet for feature extraction; applied clustering for diversity.	Achieved high accuracy (88.5%), precision (90.2%), and recall (87.1%) while maintaining recommendation relevance.	Requires significant computational resources for training and implementation.
[17] Huang et al. (2025)	Optimize personalized product recommendations by considering stochastic purchase probability.	Two-stage recommendation system; logistic regression model for user preferences and stochastic optimization for revenue maximization.	Balances recommendation accuracy with platform revenue optimization.	Stochastic modeling may not fully capture real-world purchasing behavior.
[18] Sharma &Paço (2025)	Analyze how different influences shape green purchasing behavior in e-commerce.	Used the O ZONE model and SOBC framework; PLS-SEM and SmartPLS analysis.	Provides insights into factors influencing green product awareness and self-efficacy.	Findings are based on self-reported data, which may have biases.

[19] Li et al. (2025)	Improve agricultural product recommendations using multimodal AI.	Introduced LLAVA for data enhancement; fused product association relationships; extracted user modal preferences.	Enhanced recommendation accuracy on Amazon dataset compared to baselines.	Focused on agricultural products; may not generalize to other e-commerce categories.
[20] Ahmad (2025)	Explore digital technologies' role in enhancing supply chain efficiency in e-commerce.	Qualitative study with interviews, focus groups, and document analysis.	Highlights AI, IoT, blockchain, and big data analytics' benefits in logistics and inventory management.	Lacks quantitative performance metrics for assessing impact.

Identified Research Gaps

Despite advancements in e-commerce recommendation systems, several research gaps remain. Many models, such as those by Sameena et al. [11] and Li et al. [19] lack generalizability beyond specific datasets, limiting their broader applicability. Trinh et al. [12] improved scalability but did not explore multi-model approaches to enhance accuracy, while Krishna et al. [15] highlighted the need for real-world testing of sentiment-based recommendations. Personalization challenges persist due to data privacy concerns [13], and recommendation diversity remains computationally demanding [16]. Stochastic modelling assumptions [17] and biases in consumer behaviour studies [18] further impact recommendation reliability. Additionally, Ahmad [20] identified a lack of quantitative metrics in digital supply chain studies, and domain-specific applications (e.g., agricultural products in [19]) limit broader insights. Future research should explore hybrid recommendation approaches, privacy-preserving personalization, and scalable yet accurate models for improved e-commerce experiences.

3. Methodology

Dataset: The dataset comprises three key 'csv' files as shown in Table 2: Events, which tracks user interactions (views, add-to-cart actions, and purchases) with timestamps; Item Properties, detailing product attributes like category and availability; and Category Tree, defining parent-child category relationships. Out of 1,407,580 visitors, only 11,719 completed purchases (~0.83% conversion rate), with the rest primarily browsing. Transactions are identified by non-null transaction id values, while other interactions remain NaN. A recommendation approach suggests identifying products frequently bought together to enhance user experience and conversion rates. **Preprocessing:** Before starting the simulation, training and testing the pre-processing steps are completed. This process involves the following steps:

Check for missing values in the datasets: This figure 4 shows missing values in three data sets. It is important to note that the zero in data fields does not always indicate a missing value or a default one past datasets: first data set that is shaped (1, 5) include columns such as timestamp, visitorid, event, itemid, and transactionid where all fields contain zeros except transactionid which contains a legitimate value, 2733644. Second data set shaped (1, 2) include categoryid and parentid with categoryid set as zero and parentid containing a legitimate value, 25. Last data set shaped (1, 4) include timestamp, itemid, property and value where all of them are zeros. These patterns can interpret as a potential issue of data completeness that would require further investigations to prove whether zero indicates missing values, placeholders or legitimate ones.

Table 2. *Datasets Description*

Dataset Name	Columns	Description
Events (events_df)	timestamp, visitorid, event, itemid, transactionid	Tracks user interactions (views, add to cart, purchases)
Item Properties (item_properties_1_df)	timestamp, itemid, property, value	Item attributes like category ID and stock availability
Category Tree (category_tree_df)	categoryid, parentid	Defines category relationships in a hierarchical structure

timestamp --- i64	visitorid --- i64	event --- str	itemid --- i64	transactionid --- str
1433221332117	257597	view	355908	null
1433224214164	992329	view	248676	null
1433221999827	111016	view	318965	null
1433221955914	483717	view	253185	null
1433221337106	951259	view	367447	null

categoryid --- i64	parentid --- i64
1016	213
809	169
570	9
1691	885
536	1691

Events(A)	Item Properties(B)

timestamp --- i64	itemid --- i64	property --- str	value --- str
1435460400000	460429	categoryid	1338
1441508400000	206783	888	1116713 960601 n277.200
1439089200000	395014	400	n552.000 639502 n720.000 42456…
1431226800000	59481	790	n15360.000
1431831600000	156781	917	828513

Category Tree(C)

Figure 3. *Sample records of all three datasets*

timestamp --- u32	visitorid --- u32	event --- u32	itemid --- u32	transactionid --- u32
0	0	0	0	2733644

shape: (1, 2)

categoryid --- u32	parentid --- u32
0	25

shape: (1, 4)

timestamp --- u32	itemid --- u32	property --- u32	value --- u32
0	0	0	0

Figure 4. *Missing values in the datasets*

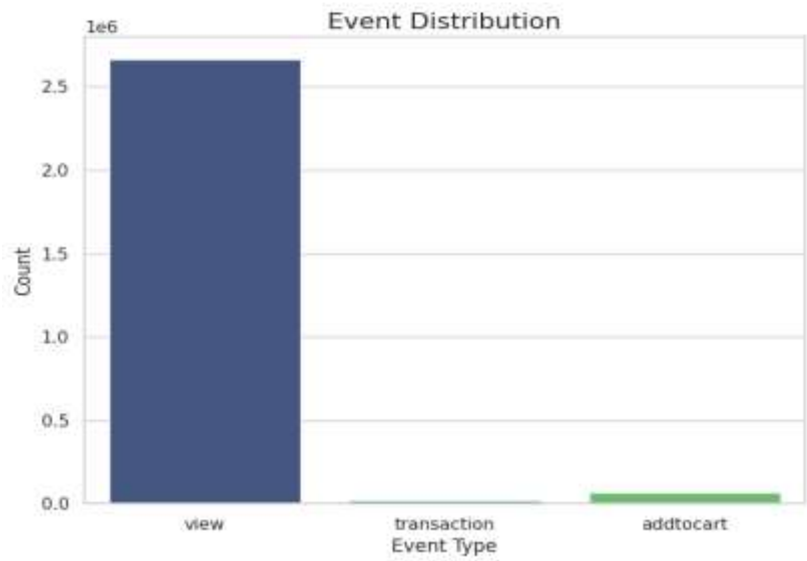

Figure 5. *Event Distribution*

Event Distribution: The bar chart in Figure 5 visualizes the structure of the dataset by different event types: view, cart and transaction events. It can be seen that the view event is the most popular, as it has a count exceeding 2.5 million, showing that the most interactions are users' viewing products. The cart and transaction events are much less frequent, which means that less than half of views goes to the cart and a smaller part goes to customers who buy. Thus, the dataset can be visualized as the funnel where view is the most frequent event, and the others are less popular. This information can be optimized to make more customers buy.

Figure 6 depicts the logarithmic scale distribution. This distribution represents the frequency of different purchase counts across users. This histogram shows most users only make a few purchases. The most common purchase count value is at the lower end. As the purchase count increased, the frequency decreased in a non-linear fashion. This indicates that very few users make a lot of purchases. There are several extremely high purchase counts, which represent a small group of highly engaged buyers. Logarithmic scale is better because it illustrates the nature of purchase behaviour. It shows that few users make most of the purchases.

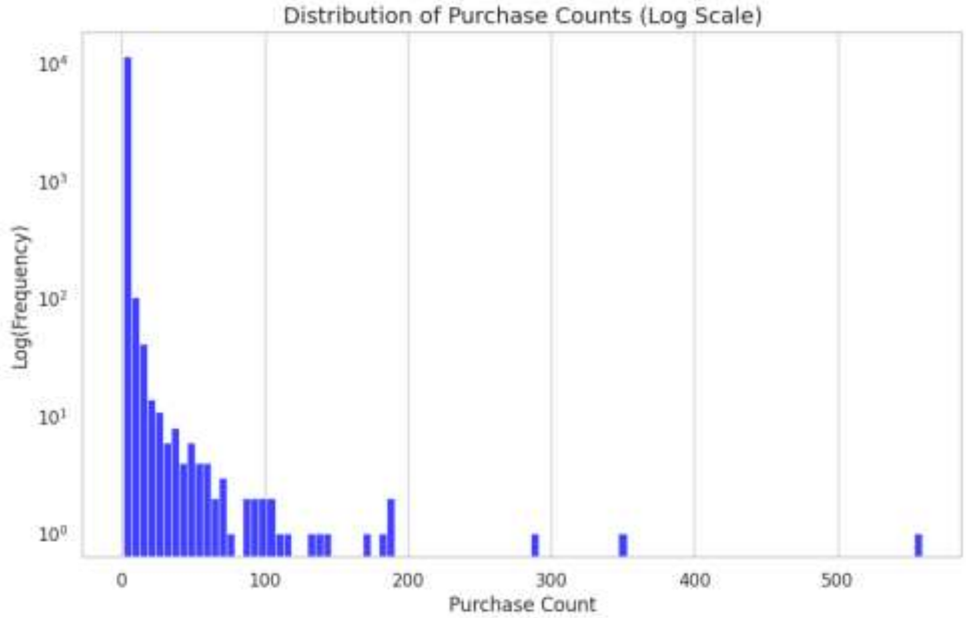

Figure 6. *Distribution of purchase counts (log Scale)*

Figure 7 displays the top 20 customers ranked by their total purchase count. The bar chart reveals that the highest purchasing customer (Visitor ID 113086) has made over 500 purchases, significantly surpassing the second-highest customer. The purchase counts decrease gradually among the top buyers, with the lower-ranked customers in this group making around 100 purchases. The colour gradient in the bars emphasizes the distribution,.

Figure 7. *Top twenty customers by purchase count*

with darker shades representing higher purchase counts. This visualization highlights the presence of high-value customers who contribute substantially to overall sales, making them crucial for retention and targeted marketing strategiesPeak Purchase Times: The Peak Purchase Times chart in figure 8 represents the allocation of similar buys among different hours of the day. The info indicates a substantial increase in buyings starting at 13:00, then reaching its peak between 14:00 and 15:00 with about 7,000 purchases, finally, decreasing after 15:00. Therefore, the findings indicate that the high online shopping activity takes place in the afternoon, perhaps, it relates to lunch time or predetermined free time. The companies are capable of utilizing the data to launch their ads and promotional campaigns, provide discounts, and market their products during these peak hours to raise more sales. The Top 10 Selling Products bar chart in figure 9 delineates the most purchased items based on

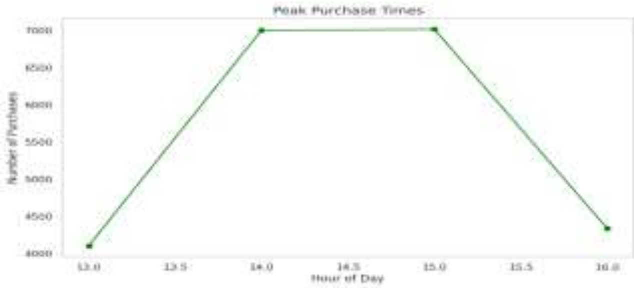

Figure 8: *Peak Purchase Times*

transaction data. Product 461686 is the most popular one with over 120 purchases, followed by 119736 and 213834, both having more than 90 purchases. The rest of the products, such as 317228, 7903, and 45531, illustrate a steady drop in sales, with the weakest one among the top 10 still managing to acquire sales. This finding will not only guide companies in selecting the most demanding products but also will teach them how to offer bestseller products more effectively and thus promoting bundle sales.

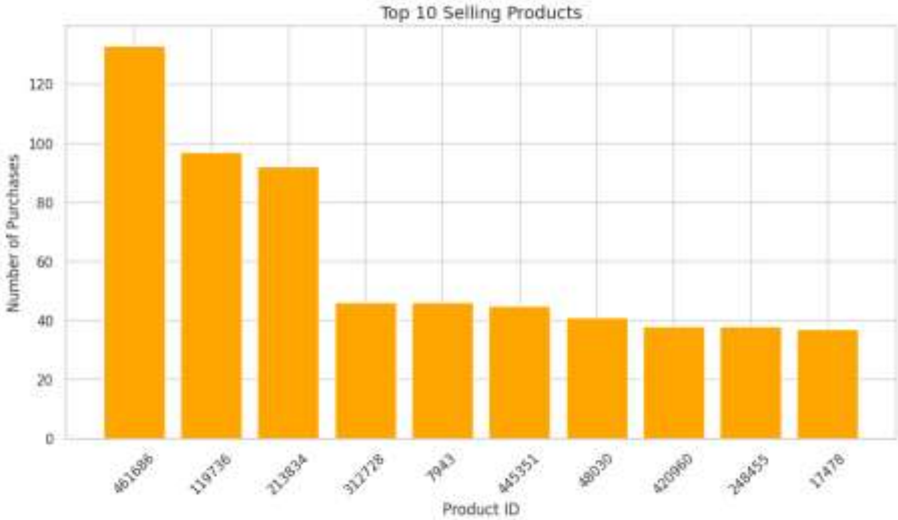

Figure 9. *Top 10 Selling products*

The visual representation in figure 10 of the top 10 most popular product categories is displayed in the bar chart that shows event counts and is likely composed of views, add-to-cart actions, and purchases. Category 1613 tops the list with almost 500,000 events, while Category 491 comes next with more than 350,000 interactions. Initiatives such as 1120, 1509, and 1277 are declared very successful by the engagement and sales, while the rest of the categories are performing well with a lower number of events. Through these insights, businesses can better manage their inventory, develop strategic marketing initiatives, and also attract the attention and enthusiasm of customers through the promotion of the most popular products.

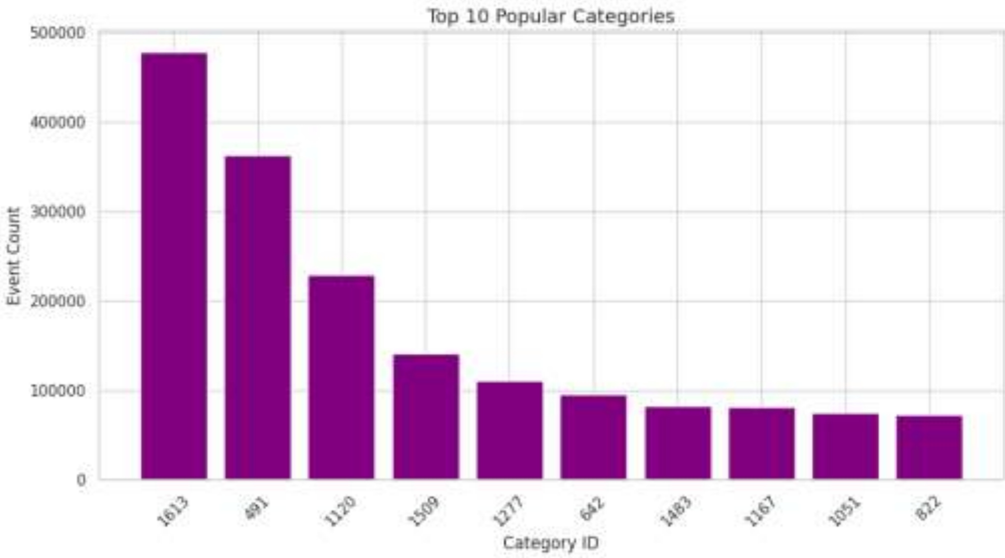

Figure 10. *Top 10 Popular categories*

The holistic recommendation system utilized three different datasets—Events (A), Item Properties (B), and Category Tree (C)—are merged over shared keys and time alignment. The Events dataset forms the base, recording user activity (e.g., views on items) along with time, visitor ID, and item ID. In order to enrich these events with item metadata, we merge the Item Properties table on the itemid field in order to introduce extra item characteristics like categoryid and custom property values (i.e., price, brand, etc.). Secondly, the categorized categoryid found in item properties is merged into the Category Tree, which returns hierarchical category info by mapping each categoryid back to its related parentid. This facilitates modelling of category-level preference and item similarity using hierarchy-conscious recommendation methods. The integrated dataset therefore reflects user-item interactions, item attributes, and the semantic item category structure, creating an affluent basis for creating content-based, collaborative, or hybrid recommendation models.

Proposed Model: The figure describes the process of an Attention-Driven Deep Learning Model for recommendation systems. The process starts with Datasets, which go through Data Pre-Processing & Analysis to get cleaned and formatted data. The datasets is then divided into Train (80%) and Test (20%) sets. The Training Data is employed for Feature Engineering, where insightful representations are extracted for learning. These characteristics are input into the Proposed Attention-Driven Deep Learning Model (PRE-ADDL), which improves user-item interactions by the introduction of an attention mechanism. The resultant Trained Model is tested with Unseen Data to verify its generalization capability. The model goes through Evaluation, where its performance on Accuracy & Recommendation Quality is determined to ensure its ability to accurately predict user preferences.

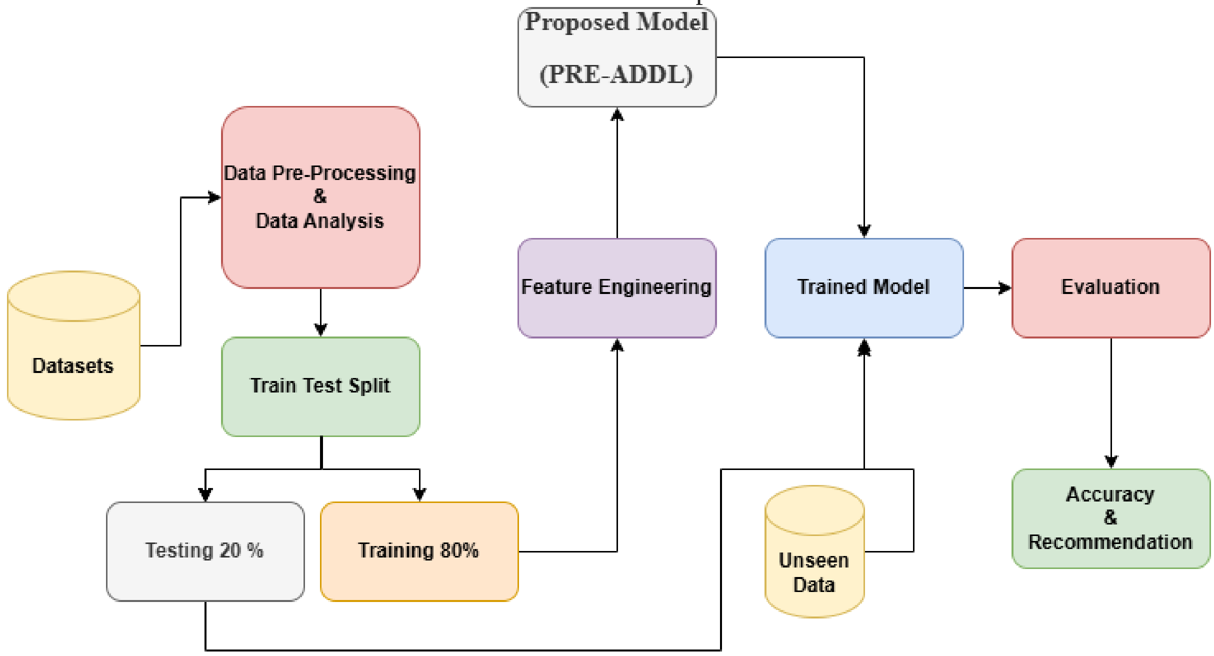

Figure 11. *Block diagram of proposed model*

Proposed Model Architecture: Layer-by-Layer description of the deep learning model shown in Figure 12:

1. Input Layers

deep_input (InputLayer) – Takes a 2-dimensional input representing visitor and item IDs.

Attention_input (InputLayer) – Another input layer for additional attention-based processing.

2. Embedding Layers (Feature Representation)

visitor_embedding (Embedding) – Maps visitor IDs to an 8-dimensional embedding space.

item_embedding (Embedding) – Maps item IDs to an 8-dimensional embedding space.

3. Flatten Layers (Reshaping the Embeddings)

flatten (Flatten) – Flattens the visitor embedding from (None, 8) to a 1D vector.

flatten_1 (Flatten) – Flattens the item embedding from (None, 8) to a 1D vector.

4. Concatenation Layer (Merging Features)

deep_concat (Concatenate) – Merges the visitor and item embeddings into a single 16-dimensional feature vector.

5. Attention Mechanism (Feature Importance Learning)

attention_layer1 (Dense) – A fully connected layer with 64 neurons, learning feature importance.

attention_layer (AttentionLayer) – Captures dependencies between user-item interactions.

dropout1_with_attention (Dropout) – Regularization to prevent overfitting in the attention layer.

6. Dense Layers (Further Feature Processing)

attention_layer2 (Dense) – A fully connected layer with 32 neurons to refine attention-based features.
dropout2 (Dropout) – Regularization applied to the 32-neuron dense layer.

7. Output Processing Layers

Attention_output (Dense) – Produces an output score from attention-based features.
drop1_with_attention (Dense) – Another dense layer generating an alternative prediction score.
AttentionLayer_merge (Add) – Combines the outputs from attention-based and standard dense layers.
final_output (Dense) – The final layer producing a single prediction value (e.g., recommendation score).

Algorithm : Outlines the steps taken in the PRE-ADDL model

This algorithm outlines the steps taken in the PRE-ADDL model, which leverages attention mechanisms and deep learning for personalized product recommendations in e-commerce.

Step 1: Data Loading and Preprocessing

- **Import Libraries:** Load necessary libraries such as NumPy, Pandas, TensorFlow, scikit-learn, and Matplotlib.
- **Load Dataset:** Read user event logs, product details, and category information from CSV files.
- **Data Cleaning:** Convert categorical variables into numerical representations using Label Encoding. Normalize numerical features using Min-Max Scaling.
- **Train-Test Split:** Split data into 80% training and 20% testing using train_test_split. Ensure a balanced distribution of positive and negative interactions.

Step 2: Model Architecture Design

- **Define Input Layers:**
  - Create embeddings for users and products using Embedding layers. Process numerical features separately using Dense layers.
  - Apply Attention Mechanism: Use an Attention Layer to weigh user-item interactions. Extract meaningful features by learning which interactions are most relevant.
- **Concatenate Features:**
  - Merge user embeddings, product embeddings, and numerical features into a unified vector.
- **Final Classification Layer:**
  - Pass the concatenated vector through Dense layers with activation functions.
  - Use a Sigmoid activation in the final layer for probability prediction.
- **Compile Model:**
  - Use Adam Optimizer for fast convergence.
  - Define Binary Cross-Entropy as the loss function.

Step 3: Model Training and Validation

- **Set Hyperparameters:** Define batch size, number of epochs, and dropout rates to prevent overfitting.
- **Train the Model:** Train using the training dataset with batch processing. Monitor validation loss to avoid overfitting.
- **Evaluate Model Performance:** Compute Confusion Matrix. Generate Classification Report (Precision, Recall, F1-Score).

Step 4: Recommendation System Deployment

- **Predict User Preferences:** Use the trained model to predict interaction probabilities for user-product pairs.
- **Generate Personalized Recommendations:** Rank products based on predicted probability scores. Filter out low-confidence recommendations.
- **Optimize Response Time:** Ensure fast inference time (~0.63 seconds per recommendation).

Mathematical Representation:

Let:
X be the input vector of shape (2,).
V and I be the visitor and item embedding matrices.

$E_v = V[x1]$ and $E_i = I[x2]$ be the embeddings.

h = Concat(E_v,E_i) be the concatenated embedding
Attention mechanism involves:

Dense transformation: $h_1 = \sigma(W_1 h + b_1)$

Attention weighting: a=Softmax($W_2 h_1 + b_2$)

Weighted sum: $h' = a.h_1$

Final prediction: $y = \sigma(W_{final} h' + b_{final})$

This mathematical model is used for recommendation systems where visitor-item interactions

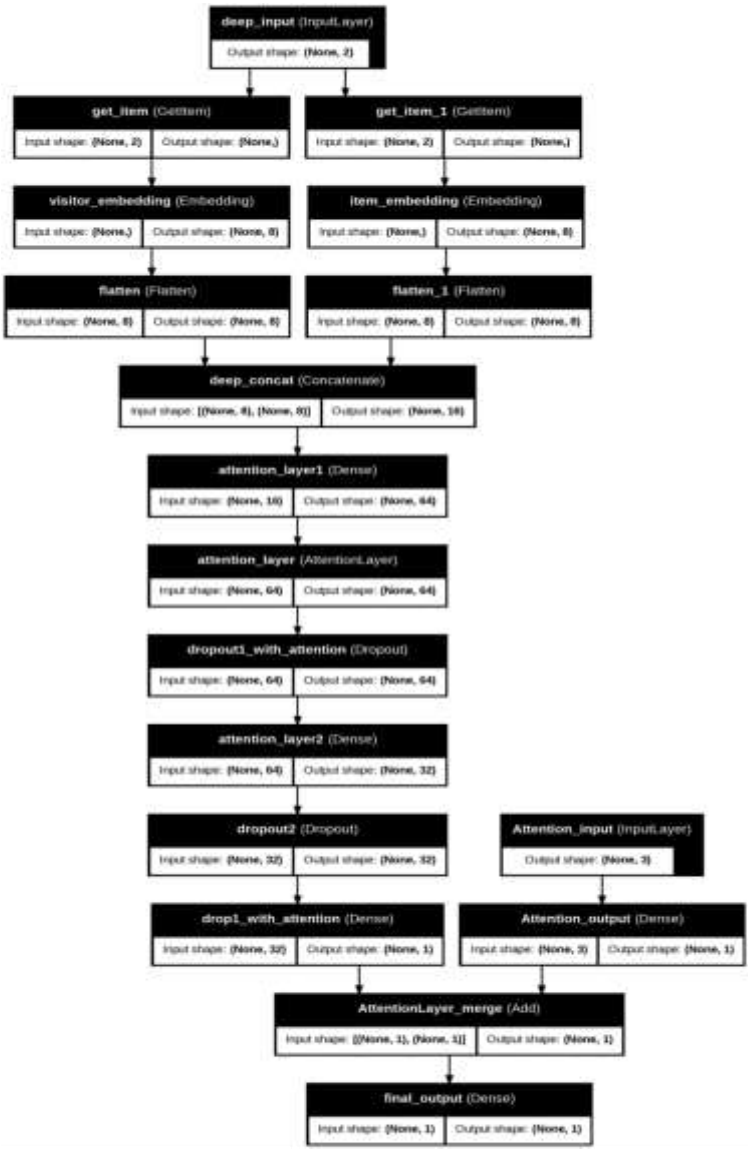

Figure 12. *Proposed (PRE-ADDL) Attention-Driven Deep Learning Model Architecture*

4. Results and Experimentation

The Proposed (PRE-ADDL) Attention-Driven Deep Learning Model reduces training time by approximately 78% compared to SVD and 97% compared to BERT, demonstrating a significant improvement in computational efficiency over the other models. The testing time was lowered by approximately 16% in comparison to the Naïve Bayes Classifier. The recommendation response time also decreases from 7 minutes to just 3 minutes, which is about 57% faster than BERT. This makes the system more scalable and responsive for real-time applications. These improvements demonstrate how the suggested model can provide high-performance suggestions while drastically cutting down on computational overhead, making it a sensible and effective option for real-world implementation.

Table 3. *Overview of computational efficiency for different algorithms*

Models	**Single Epochs training time**	**Single Epochs Test time**	**Recommendation Response**
BERT Model [22]	78 hours	48 hours	7 minutes
(PRE-ADDL)	1 hour 40 minutes	50 minutes	3 minutes
SVM	7 hours 43 minutes	57 minutes	8 minutes
K-Nearest Neighbor	3 hours 32 minutes	55 minutes	6 minutes
Naïve Bayes Classifier	5 hours 22 minutes	1 hours	5 minutes
Decision Tree	4 hours 23 minutes	53 minutes	4 minutes

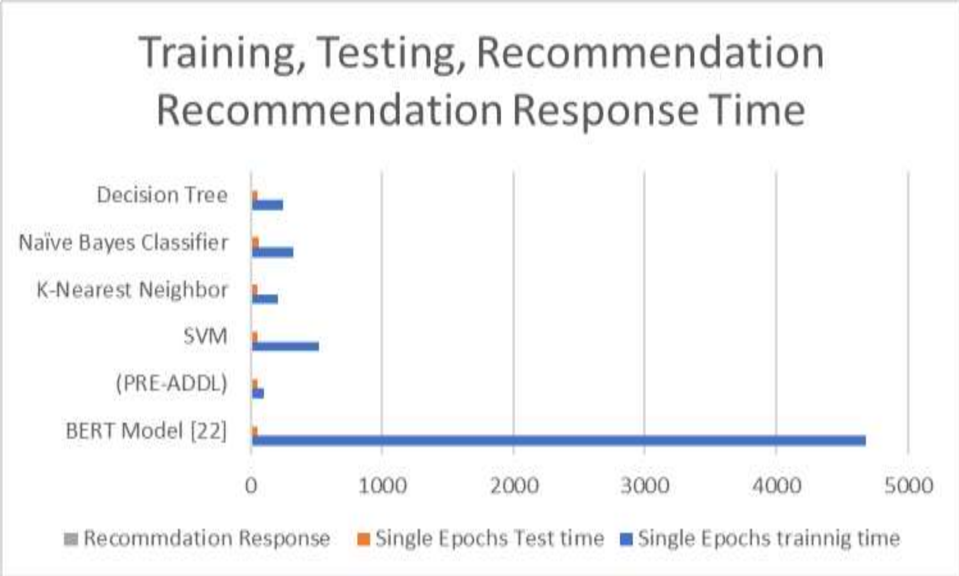

Figure 13. *Training, Testing, and Recommendation Response Timing*

Table 4 provides performance measures of different models based on important factors like Accuracy, F1-score, Precision, and Recall. The findings show serious differences in the performances of different models.

The PRE-ADDL model shows the best accuracy 0.97, and F1-Score 0.98 reflects its better capacity to accurately predict instances while ensuring precision and recall are within a healthy balance. The precision score 0.93 reflects strong confidence in optimistic predictions, and the recall score 0.97 validates its resilience to predict actual positive instances. BERT model comes next with a precision of 0.79 and an F1-Score of 0.88. Although its recall 0.97 is high, meaning that it is good at picking positive instances, its precision 0.81 is lower, meaning that there are occasional misclassifications. The Support Vector Machine (SVM) model has an accuracy of 0.75, F1-Score of 0.86, and precision of 0.77. Even though it has relatively lower accuracy, its recall value of 0.97 is high, which makes it efficient in the detection of true positive cases. The K-Nearest Neighbor (KNN) model and the Naïve Bayes Classifier register fairly poor performance, with accuracy values of 0.65 and 0.61, respectively. Their F1-Scores of 0.78 and 0.75 reflect that these models have poor precision, garnering values of 0.66 and 0.62, respectively. The two models register high recall scores 0.97 and 0.96, reflecting that they tend to identify positive instances but at the expense of higher false positives. The Decision Tree model also shows a moderate performance with an accuracy of 0.74 and F1-Score 0.84. Although its precision 0.75 is not as good as some other models, its recall 0.97 is still high, showing that it can effectively identify most of the positive instances.

Table 4. *Performance measures of different models*

Model	Accuracy	F1-Score	Precision	Recall
PRE-ADDL	0.9665	0.983	0.100	0.9665
BERT	0.7956	0.8846	0.81	09742
SVM	0.759	0.8609	0.7714	0.9738
K-Nearest Neighbor	0.6511	0.7863	0.663	0.966
Naïve Bayes Classifier	0.6179	0.7595	0.6242	0.9696
Decision Tree	0.7427	0.8497	0.7526	0.9756

The proposed model learning capability is essential to achieve higher accuracy and better generalization. This involves optimizing the learning rate, adjusting network depth, utilizing attention mechanisms, and incorporating hybrid architectures. The learning rate plays a crucial role in controlling how quickly a model updates its weight. A well-tuned learning rate ensures the model converges efficiently. The learning rate schedule helps adjust learning dynamics over training epochs. The integration of attention mechanisms with CNNs, RNNs, or transformers enhances learning capability by selectively focusing on essential information, improving feature representations, and reducing unnecessary computations, optimal balanced updates, efficient feature learning, faster convergence, and better generalization.

5. **Conclusion and Future Work**

This work effectively deployed an Attention-Driven Deep Learning Model (PRE-ADDL) for product personalization in e-commerce. From the analysis of user interactions, buying patterns, and event distributions, insights about consumer behavior showed that view events prevail but have low conversion rates. With the aid of deep learning and attention-based mechanisms, the model identified meaningful user-item interactions well, achieving approximately 97% classification accuracy. The recommendation system effectively offered suggestions for similar products with fast computation time (0.63 seconds). Future research can be directed toward enhancing personalization through the use of hybrid recommendation strategies, including collaborative filtering and reinforcement learning, to maximize outcomes. Tackling the skewed class distribution and investigating real-time adaptation for changing user preferences can also improve recommendation performance. The introduced model provides a strong framework for personalized recommendations with important implications for enhancing customer engagement and e-commerce sales performance. In future studies, it is possible to enhance by hybrid recommendation techniques, merging collaborative filtering and reinforcement learning to further personalize. Dealing with the imbalanced class distribution in user behavior data and investigating real-time adaptive processes for dynamically changing preferences can contribute significantly to system performance. These improvements will not only enhance recommendation accuracy and user engagement rates but also make the system scalable and adaptable in dynamic e-commerce settings. The model proposed creates a strong foundation for sophisticated personalization methods, which leads to better customer experience and higher sales conversion in online shopping platforms.

Author Statements:

- **Ethical approval:** The conducted research is not related to either human or animal use.
- **Conflict of interest:** The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper

- **Acknowledgement:** The authors declare that they have nobody or no-company to acknowledge.

- **Author contributions:** The authors declare that they have equal right on this paper.

- **Funding information:** The authors declare that there is no funding to be acknowledged.

- **Data availability statement:** The data that support the findings of this study are available on request from the corresponding author. The data are not publicly available due to privacy or ethical restrictions.

References

[1] Cabrera-Sánchez, J. P., Ramos-de-Luna, I., Carvajal-Trujillo, E., & Villarejo-Ramos, Á. F. (2020). Online recommendation systems: Factors influencing use in e-commerce. *Sustainability*, 12(21),

[2] Lin, Z. (2014). An empirical investigation of user and system recommendations in e-commerce. *Decision Support Systems*, 68, 111-124.

[3] Tran, D. T., & Huh, J. H. (2023). New machine learning model based on the time factor for e-commerce recommendation systems. *The Journal of Supercomputing*, 79(6), 6756-6801.

[4] Li, S. S., & Karahanna, E. (2015). Online recommendation systems in a B2C E-commerce context: a review and future directions. *Journal of the Association for Information Systems*, 16(2), 2.

[5] Srivastava, J., Cooley, R., Deshpande, M., & Tan, P. N. (2000). Web usage mining: Discovery and applications of usage patterns from web data. *ACM SIGKDD Explorations Newsletter*, 1(2), 12-23.

[6] Addagarla, S. K., & Amalanathan, A. (2020). Probabilistic unsupervised machine learning approach for a similar image recommender system for E-commerce. *Symmetry*, 12(11), 1783.

[7] Zhu, Z., Wang, S., Wang, F., & Tu, Z. (2022). Recommendation networks of homogeneous products on an E-commerce platform: Measurement and competition effects. *Expert Systems with Applications*, 201, 117128.

[8] Louca, R., Bhattacharya, M., Hu, D., & Hong, L. (2019). Joint optimization of profit and relevance for recommendation systems in e-commerce. *RMSE@ RecSys*, 2440, 4-7.

[9] Jiang, L., Cheng, Y., Yang, L., Li, J., Yan, H., & Wang, X. (2019). A trust-based collaborative filtering algorithm for E-commerce recommendation system. *Journal of Ambient Intelligence and Humanized Computing*, 10, 3023-3034.

[10] Mykhalchuk, T., Zatonatska, T., Dluhopolskyi, O., Zhukovska, A., Dluhopolska, T., & Liakhovych, L. (2021, September). Development of recommendation system in e-commerce using emotional analysis and machine learning methods. In 2021 *11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS) (Vol. 1, pp. 527-535). IEEE.*

[11] Sameena, S., Javali, G., Srilakshmi, N., Jhansi, M., & Sk, S. S. (2025). Personalized product

recommendation system for e-commerce platforms. In ITM Web of Conferences (Vol. 74, p. 03012). *EDP Sciences.*

[12] Trinh, T., Nguyen, V. H., Nguyen, N., & Nguyen, D. N. (2025). Product collaborative filtering-based recommendation systems for large-scale E-commerce. *International Journal of Information Management Data Insights*, 5(1), 100322.

[13] Vashishth, T. K., Sharma, K. K., Kumar, B., Chaudhary, S., & Panwar, R. (2025). Enhancing customer experience through AI-enabled content personalization in e-commerce marketing. *Advances in Digital Marketing in the Era of Artificial Intelligence*, 7-32.

[14] Wu, W., Li, X., Wang, L., Zhou, J., Wu, D., Xie, Q., ... & Chen, J. (2025). IU4Rec: Interest Unit-Based Product Organization and Recommendation for E-Commerce Platform. *arXiv preprint* arXiv:2502.07658.

[15] Krishna, E. P., Ramu, T. B., Chaitanya, R. K., Ram, M. S., Balayesu, N., Gandikota, H. P., & Jagadesh, B. N. (2025). Enhancing E-commerce recommendations with sentiment analysis using MLA-EDTCNet and collaborative filtering. *Scientific Reports*, 15(1), 6739.

[16] Bahi, A., Gasmi, I., Bentrad, S., & Khantouchi, R. (2025). Enhancing recommendation diversity in e-commerce using siamese network and cluster-based technique. *Bulletin of Electrical Engineering and Informatics*, 14(2), 1223-1230.

[17] Huang, C., Zhang, X., Zhang, Y., & Hu, Q. (2025). Revenue-based personalized product recommendation considering stochastic purchase probability. *Electronic Commerce Research and Applications*, 70, 101477.

[18] Sharma, N., & Paço, A. (2025). Exploring the O Zone effect: how own, others and outer influences shape green purchasing behaviour in e-commerce. *Journal of Information, Communication and Ethics in Society*.

[19] Li, P., Gao, L., Zhang, L., Peng, L., Bai, C., & Yang, L. (2025). MP-LLAVRec: an agricultural product recommendation algorithm based on LLAVA and user modal preference. *Frontiers in Physics*, 13, 1525353.

[20] Ahmad, N. R. (2025). Exploring the Role of Digital Technologies in Enhancing Supply Chain Efficiency: A Case Study of E-Commerce Companies. *Indus Journal of Social Sciences*, 3(1), 226-237.

[21] Pan, L. (2025). Hybrid collaborative recommendation of cross-border e-commerce products based on multidimensional evaluation. International *Journal of Information and Communication Technology*, 26(1), 102-116.

[22] Karabila, I., Darraz, N., EL-Ansari, A., Alami, N., & EL Mallahi, M. (2024). BERT-enhanced sentiment analysis for personalized e-commerce recommendations. *Multimedia Tools and Applications*, 83(19), 56463-56488.