International Journal of Emerging Research in Engineering, Science, and Management
Vol. 5 | Issue 3 | pp. 44-54 | Jul-Sep 2026.
www.ijeresm.com | eISSN – 2583-4894
Received: 14 Mar 2026 | Revised: 16 Jun 2026 | Accepted: 22 Jun 2026 | Published: 22 Jul 2026

Development of an Interactive Web-Based Hybrid E-Commerce Recommendation System Using Neural Collaborative Filtering and Content-Based Filtering

[1]Muhammad Farid, [2]Tasyaul Husna, [3]Muhammad Zia Ulhaq, [4]Teuku Muhammad Faiz Nuzullah, [5]Teuku Rizky Noviandy, [6]Novi Reandy Sasmita

[1-4,6]Department of Statistics, Faculty of Mathematics and Natural Sciences, Universitas Syiah Kuala, Banda Aceh, Indonesia.
[5]Department of Information Systems, Faculty of Engineering, Universitas Abulyatama, Aceh Besar, Indonesia.

[1]mhdfaridz93@gmail.com, ORCID iD: 0009-0001-6436-1396,
[2]husna22@mhs.usk.ac.id, ORCID iD: 0009-0007-6584-2283,
[3]m_zia@mhs.usk.ac.id, ORCID iD: 0009-0004-7812-8095,
[4]faiznuzullah@gmail.com, ORCID iD: 0009-0000-0761-9525,
[5]rizky_si@abulyatama.ac.id, ORCID iD: 0000-0002-5779-2235,
[6]novireandys@usk.ac.id, ORCID iD: 0000-0002-2840-0335.

**Corresponding author: Novi Reandy Sasmita*

Abstract: In an increasingly digital era, technological advancements have transformed the way people shop and interact with e-commerce platforms. E-commerce offers a wide range of products and services online, but consumers often feel overwhelmed by the sheer number of choices. The importance of efficient and relevant services for consumers is a significant consideration in developing e-commerce systems. One effective solution is implementing a recommendation system that helps consumers find products that match their needs and preferences. This study develops an e-commerce recommendation system using neural collaborative filtering and content-based filtering methods, integrated with an interactive dashboard to improve user experience. With a learning rate of 0.01 and a batch size of 32, the developed neural collaborative filtering model shows a low testing error of 0.0705 and precision, recall, and F-measure of 0.9583, 0.8251, and 0.8867, respectively. The content-based filtering method uses the Cosine Similarity technique, achieving an Average Precision of 0.804 and an accuracy of 80%. The interactive dashboard was successfully developed to support system evaluation and user engagement. This study provides benefits for consumers who can find products according to their needs more quickly and for e-commerce business owners who can reach consumers more efficiently.

Keywords: Recommendation system, Content-based filtering, Hybrid recommendation system, Neural collaborative filtering, E-commerce.

1 Introduction

In the digital era, transforming traditional commerce into e-commerce has revolutionized consumer behavior and business operations. With an expansive selection of online products and services, consumers often face challenges in identifying items that align with their preferences and needs. The proliferation of e-commerce platforms has enabled businesses to expand their market reach and foster greater competition in the global digital economy. According to the study by Nazir and Roomi [1], e-commerce empowers businesses, especially small and medium enterprises, to improve efficiency and engage with geographically dispersed customers. In Indonesia, the e-commerce user base increased by 12.79% in 2022, reaching 158.65 million users, with projections estimating an additional 33.5 million users by 2029 [2]. This growth underscores the pressing need for practical tools to help consumers navigate vast product catalogs and make informed purchasing decisions.

To address this challenge, recommendation systems have emerged as essential components of modern e-commerce platforms. These systems analyze user data, such as purchase history, ratings, and preferences, to generate relevant product suggestions, thereby enhancing user satisfaction and accelerating decision-making [3]. Numerous studies have highlighted the dual benefits of recommendation systems: improving consumer experiences and helping businesses reach their target audiences more effectively [4]. However, despite their growing popularity, conventional recommendation systems often struggle with limitations such as data sparsity, cold-start problems, and lack of scalability.

International Journal of Emerging Research in Engineering, Science, and Management
Vol. 5 | Issue 3 | pp. 44-54 | Jul-Sep 2026.
www.ijeresm.com | eISSN – 2583-4894
Received: 14 Mar 2026 | Revised: 16 Jun 2026 | Accepted: 22 Jun 2026 | Published: 22 Jul 2026

To address these challenges, recent literature has focused on hybrid methods that combine content-based and collaborative filtering and leverage big data and deep learning technologies. Content-based filtering and collaborative filtering are two widely adopted approaches in recommendation systems. The former suggests items based on the similarity between a user's past interactions and product attributes, while the latter relies on similar users' preferences. Although several studies have explored these techniques independently, integrating them into a hybrid model has demonstrated promising improvements in recommendation quality. For instance, Sami et al. [5] developed a hybrid model that incorporates content-based filtering using TF-IDF and cosine similarity to recommend items based on specific attributes, such as user-defined genres, while Yun et al. [6] highlighted collaborative filtering's potential to leverage user reviews for more personalized recommendations. While recent hybrid models have significantly improved accuracy, they often operate as black-box systems, providing results without an accessible interface for stakeholders to interpret the decision-making process behind the recommendations.

Recent advancements in Artificial Neural Networks (ANN) and deep learning have introduced a transformative approach to collaborative filtering, enabling systems to model complex nonlinear relationships in large datasets. Composed of interconnected layers of neurons, these neural models have shown superior performance in predictive tasks, including recommendation systems. Specifically, Hiroki et al. [7] demonstrated the effectiveness of neural collaborative filtering in handling both sparse and abundant multicriteria evaluation data, directly addressing the persistent challenge of data sparsity while enhancing predictive accuracy. Furthermore, recent literature has highlighted that deep learning approaches can significantly outperform conventional techniques. For instance, an efficient deep learning approach for collaborative filtering proposed in recent studies [8] overcomes the computational challenges and limitations of traditional methods such as matrix factorization and K-nearest neighbor (KNN) in capturing deep behavioral patterns. Integrating these advanced neural architectures with collaborative filtering thus presents a highly effective approach to address the shortcomings of existing systems.

However, effective recommendation systems require accurate prediction algorithms and user-centric visualization tools. Interactive dashboards serve as an effective interface for presenting recommendation outcomes, enabling users to explore personalized suggestions in real time and providing business stakeholders with actionable insights into consumer behavior. Tschinkel demonstrated the usefulness of recommendation dashboards for visualizing recommendation outputs and improving user interaction with recommendation systems [9]. However, the integration of ANN-based collaborative filtering with interactive dashboards in e-commerce remains relatively underexplored. Previous studies attempted to enhance recommendation systems using machine learning and hybrid techniques. Jayalakshmi et al. used K-Means clustering and KNN for movie recommendations but encountered limitations due to high computational complexity and sensitivity to data sparsity [10]. Similarly, systems based solely on traditional algorithms often require explicit rating data, limiting their applicability in implicit feedback scenarios. These shortcomings suggest a need for more adaptive, scalable, and interpretable solutions, particularly in large-scale e-commerce environments where data heterogeneity is prevalent.

The present study addresses these challenges by proposing a hybrid recommendation system model that combines content-based filtering and neural collaborative filtering. This system is integrated into an interactive dashboard to deliver real-time, personalized recommendations to users. The proposed model processes e-commerce data to provide accurate predictions and dynamic visualizations. The integration of ANN helps overcome the sparsity issue often encountered in collaborative filtering, while the interactive dashboard bridges the gap between algorithmic complexity and user accessibility. This study makes several contributions to the field. First, it demonstrates the application of ANN to collaborative filtering in a hybrid e-commerce recommendation system. Second, it integrates an interactive dashboard as a key visualization tool, offering insights into user preferences and system performance. Third, the system supports consumers by providing more relevant recommendations and assists businesses in understanding consumer trends and behaviors.

Closely related literature includes hybrid recommendation models using deep learning and data analytics. For instance, recent architectures have utilized Transformer-based and graph-based recommender architectures to improve recommendation performance, particularly in modelling complex user-item relationships and sequential interactions [11]. However, these complex models frequently require substantial computational resources and lack seamless integration with real-time end-user interfaces. Few studies incorporate a neural-based filtering approach that balances algorithmic depth and dashboard responsiveness. Combining these technologies in a single system highlights a research gap in existing literature, where most models either focus solely on algorithmic improvements or lack interactive visualization capabilities. This study addresses this gap by offering a lightweight yet robust ANN-NCF framework directly coupled with an interactive dashboard. Therefore, this study aims to develop an interactive dashboard-based recommendation system that utilizes hybrid methods, specifically content-based filtering and neural collaborative filtering, to improve recommendation accuracy and usability.

The novelty of this study lies in integrating neural collaborative filtering with content-based filtering and developing an interactive dashboard to enhance user experience. This approach is expected to improve system performance and end-user satisfaction, offering a comprehensive solution that bridges data analytics, artificial intelligence, and user interface design in e-commerce environments. This study aims to achieve high accuracy by applying optimal hyperparameter combinations to develop a hybrid recommendation system. It is then integrated into an interactive dashboard to help consumers efficiently find products that match their preferences in an e-commerce setting.

This study helps consumers find suitable products more efficiently, enhancing their shopping experience. For e-commerce businesses, it supports product targeting and operational efficiency. It also provides a foundation for future research and technological development and contributes to inclusive and sustainable digital economic growth. This study presents an integrated framework that combines Neural Collaborative Filtering, content-based filtering, and an interactive visualization dashboard. By connecting predictive modeling with user-focused interpretability, the system delivers accurate recommendations and actionable insights for decision-making. Deployment in a Streamlit-based environment further improves accessibility and practical use in real-world e-commerce applications.

2 THEORETICAL REFERENCE

This section discusses basic concepts and prior research relevant to recommendation systems, e-commerce, and the implementation of interactive technology. Recommender systems play an important role in modern e-commerce platforms, especially in helping consumers quickly and efficiently find products that match their preferences. Recommender systems are generally divided into two main approaches, namely content-based filtering and collaborative filtering. Content-based filtering suggests products based on their similarity to products previously liked or purchased by users. In this approach, cosine similarity is often used to measure the similarity between items in the recommendation system [12]. This approach relies heavily on existing product descriptions or features. Meanwhile, collaborative filtering relies on information from the behavior of other users with similar preferences. User reviews and ratings are often used to improve the quality of recommendations in this approach. Collaborative filtering is generally divided into two types: user-based and item-based, both of which aim to find similarities in preferences between users or items based on user interaction data [5], [6].

However, both approaches often face several challenges, such as data sparsity, cold-start problems, and scalability issues. Data sparsity occurs when there is little interaction data between users and products, making it difficult to find useful patterns for recommendations. The cold-start problem occurs when the system must provide recommendations for new users or products with insufficient historical data. To overcome these problems, hybrid methods that combine the strengths of content-based filtering and collaborative filtering have been widely applied to address the limitations of each approach [13]. Alongside technological developments, several recent studies have begun integrating ANN into recommendation systems. The use of ANN enables the modeling of more complex relationships in user data, such as non-linear interactions between user preferences and the products offered. ANN can help alleviate data sparsity problems and improve prediction accuracy in recommendation systems compared to traditional methods such as matrix factorization and KNN [14].

In addition to algorithmic accuracy, the visualization of recommendation results is an important aspect of improving user experience. The use of interactive dashboards that allow users to explore recommendation results can improve users' understanding of the recommendation system and provide businesses with better insights. Good visualization can make complex data more accessible and understandable to users, thereby increasing user interaction and satisfaction with the recommendation system [15]. Recent developments also highlight the importance of deploying recommendation systems through web-based platforms to increase accessibility and scalability. Streamlit, a Python-based framework, has emerged as a popular tool for building and deploying interactive dashboards due to its simplicity and flexibility in integrating data visualization and machine learning models [16]. By providing more interpretable recommendation outputs, interactive interfaces can improve perceived transparency and user satisfaction, while also supporting more informed decision-making in e-commerce settings [17].

3 MATERIALS AND METHOD

This study proposes the development of a product recommendation system model for e-commerce platforms, integrated with an interactive dashboard. The recommendation system combines two primary approaches: neural collaborative filtering and content-based filtering. The implementation of neural collaborative filtering is intended to enhance prediction accuracy by capturing complex and non-linear patterns in user interaction data. In addition to algorithmic improvements, this study also incorporates an interactive dashboard designed to support both customers and business stakeholders. The dashboard facilitates users in discovering products aligned with their preferences and assists business owners in analyzing consumer behavior more effectively, enabling informed decisions in marketing and product strategies.

The overall methodological framework of this study is illustrated in Fig. 1. As shown in the flowchart, the research is conducted through a structured six-stage process. First, the data input stage involves collecting the Amazon Sales Dataset. Second, in the data cleaning stage, the raw data is refined by handling missing values (NaN) in the rating_count variable and processing 16 data attributes to ensure data quality. Third, Exploratory Data Analysis is performed to identify user-item interaction patterns and dataset sparsity. Fourth, building a recommendation system model is the core technical phase where a hybrid approach is developed. This involves the Neural Collaborative Filtering path, which uses deep latent layers, and the Content-Based Filtering path, which uses product attributes. Fifth, the model evaluation stage uses classification metrics (Precision, Recall, and F-Measure) to validate the recommendation accuracy. Finally, the model deployment stage concludes with the creation of a recommendation system dashboard using Streamlit, providing an interactive interface for users.

International Journal of Emerging Research in Engineering, Science, and Management
Vol. 5 | Issue 3 | pp. 44-54 | Jul-Sep 2026.
www.ijeresm.com | eISSN – 2583-4894
Received: 14 Mar 2026 | Revised: 16 Jun 2026 | Accepted: 22 Jun 2026 | Published: 22 Jul 2026

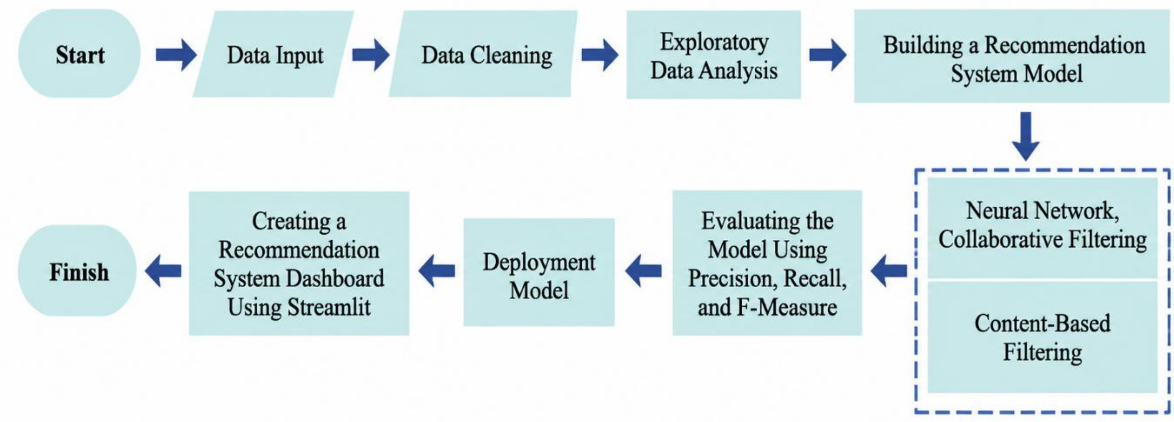

Fig. 1. Block diagram of the proposed method

The integration of these two approaches into a hybrid strategy is designed to overcome the inherent limitations of individual models. Content-Based Filtering helps address the cold-start problem by using product attributes to recommend new items with insufficient interaction history. Simultaneously, Neural Collaborative Filtering (NCF) helps overcome the limitations of Content-Based Filtering by learning complex, non-linear relationships from existing user-item interactions through its deep neural layers. This hybridization enables the system to maintain high recommendation relevance even in scenarios of high data sparsity, providing more robust and accurate performance compared to using NCF or Content-Based Filtering in isolation. This dataset was released/updated in 2023 and was obtained from the Amazon Open Data Registry (https://registry.opendata.aws/). It is beneficial for understanding market dynamics and consumer preferences. In this study, product names, reviews, and descriptions are used to match products to user preferences via content-based filtering. In contrast, in collaborative filtering, attributes such as user ID, product ID, and rating score are used to analyze user interaction patterns with products and recommend products based on similar user preferences. Products are recommended based on similarities in user preferences and interaction patterns identified from the interaction data. This study is limited to sales data and user interactions with products available on the Amazon e-commerce platform, so the results reflect user preferences based on historical platform data.

3.1. Data Preprocessing

Data preprocessing is an important step to ensure data quality before further analysis [18]. The process begins with data cleaning, which removes unwanted characters, such as currency symbols and commas, from the price column and converts the data type to the appropriate format. Next, missing values were handled to avoid bias and maintain data integrity [19]. This study uses the Amazon Sales Dataset, which comprises 16 variables, including product identifiers, user information, and rating metrics. During the initial assessment, the dataset contained 1,465 entries. Analysis of the data integrity revealed that while most variables were complete, the rating_count variable contained two missing values at indices 282 and 324. The two missing entries in the rating_count variable were removed during preprocessing because they represented only a negligible proportion of the dataset. These missing values were handled to avoid bias and maintain data integrity. The resulting clean dataset features 1,191 unique users and 1,334 products, with a total of 1,465 interactions. Once the data was cleaned, the Exploratory Data Analysis stage was conducted to identify patterns, detect anomalies, and gain initial insights through descriptive statistics and visualizations [20].

3.2. Content-Based Filtering

A content-based filtering method is used to recommend products based on the product's characteristics. This process starts with feature engineering by concatenating the product name, product reviews, and product descriptions into a single textual feature named *"Product.Detail"*. After that, the data is preprocessed by cleaning the text and applying Term Frequency-Inverse Document Frequency (TF-IDF) weighting to assign higher weights to informative terms. The TF-IDF vectorization was performed using the default Scikit-learn configuration with unigram tokenization and without additional restrictions on document frequency or vocabulary size. TF represents the frequency of a term within a product description. Meanwhile, the IDF value is determined using the following formula [21]. The text preprocessing stage includes conversion to lowercase, removal of punctuation, and stop-word removal before applying the TF-IDF transformation. Stemming and lemmatization were intentionally omitted to preserve the contextual meaning of product and brand names.

$$\mathrm{W}_{i,j} = \begin{cases} 1 + log_{10}\, tf_{t,d} & if\ tf_{t,d} > 0 \\ 0 & otherwise \end{cases} \tag{1}$$

$$\mathrm{IDF} = log\left(\frac{n}{df_i}\right) \tag{2}$$

International Journal of Emerging Research in Engineering, Science, and Management
Vol. 5 | Issue 3 | pp. 44-54 | Jul-Sep 2026.
www.ijeresm.com | eISSN – 2583-4894
Received: 14 Mar 2026 | Revised: 16 Jun 2026 | Accepted: 22 Jun 2026 | Published: 22 Jul 2026

where 'n' represents the total number of product descriptions, while df_i refers to the number of product descriptions containing term 'i'. Next, the weighted terms are used to calculate the degree of similarity between products using cosine similarity, which measures the similarity between two text vectors [22]. The formula for Cosine Similarity calculation is as follows.

$$SIM(\vec{t}_a, \vec{t}_b) = \frac{\vec{\boldsymbol{t}}_{\boldsymbol{a}} \cdot \vec{\boldsymbol{t}}_{\boldsymbol{b}}}{|\vec{\boldsymbol{t}}_{\boldsymbol{a}}| x |\vec{\boldsymbol{t}}_{\boldsymbol{b}}|} = \frac{\sum_{i=1}^{n} \vec{t}_{i,a} \vec{t}_{i,b}}{\sqrt{\sum_{i=1}^{n} \vec{t}^{2}_{i,a}} \sqrt{\sum_{i=1}^{n} \vec{t}^{2}_{i,b}}} \tag{3}$$

where $\vec{t}_a$ and $\vec{t}_b$ are multi-dimensional vectors of term sets $T = t_1, t_2, \ldots, t_n$,each dimension in the vector reflects a term with its non-negative weight in the document. Cosine Similarity always has a non-negative value and ranges from zero to one [23]. The cosine similarity matrix was computed using the cosine_similarity() function available in the Scikit-learn library. As shown in (3), the result is a similarity value that shows how similar one product is to another product [24].

3.3. Neural Collaborative Filtering

Neural Collaborative Filtering (NCF) is a technique in recommendation systems that utilizes neural networks to model the complex relationship between users and items. In NCF, embedding vectors for users and items are learned through a deep learning process, enabling the model to capture the non-linear relationships between them, resulting in more accurate and personalized recommendations [25]. This technique has been shown to be more effective in handling large and complex data sets than traditional approaches such as matrix factorization and nearest neighbors [26]. The process begins with creating the embedding layer, where the number of unique users and items is identified, and embedding vectors are generated for each. Next, the embedding vectors of both users and items are flattened and concatenated into a single vector. This vector is passed to the first dense layer, which consists of 128 neurons and uses a ReLU activation function to learn interactions between users and items. After the model is constructed, the dataset is split into 80% for training and 20% for testing.

The training data is used to help the model learn interaction patterns between users and items, while the testing data is used to evaluate its predictive accuracy. A fixed random seed of 42 was used during data partitioning to ensure reproducibility. The embedding process converts users and items into low-dimensional vectors representing their preferences and characteristics. This allows the neural network to capture complex relationships later used to predict ratings and recommend items [27]. An illustration of the neural collaborative filtering model formation scheme is shown in Figure 2 below. As shown in Figure 2, the NCF model formation involves embedding layers, followed by concatenating the user and item vectors into a single vector, which is then passed through deeper neural network layers. ReLU activation functions enable the model to capture non-linear relationships, leading to an output layer that predicts the likelihood of user-item interaction for recommendation [28]. Finally, the model is optimized using the training and testing data split. The training data teaches the model about interactions between users and items, while the testing data assesses the accuracy of its predictions [29].

The embedding dimension was selected empirically after preliminary experiments balancing predictive performance and model complexity. The NCF framework allows the model to learn the interaction between user and item embeddings through multiple layers of neural networks. To ensure reproducibility and optimal performance, the neural architecture is configured with an embedding dimension of 50 for both users and items. The model is trained with the Adam optimizer at a learning rate of 0.01 and uses Binary Cross-Entropy as the loss function. The training process consists of 20 epochs with a batch size of 32. The number of epochs was selected based on the observed convergence behaviour of the training loss. To enhance generalization and prevent overfitting, a Dropout regularization layer with a rate of 0.2 is integrated into the architecture.

The prediction for a user u on an item i is formulated as a neural interaction through the Multi-Layer Perceptron. The predicted score $\hat{y}_{ui}$ is calculated as follows:

$$\hat{y}_{ui} = \sigma\left(h^T \phi(P^T v_u^U, Q^T v_i^I)\right) \tag{4}$$

where v_u^U and v_i^I are the latent embedding vectors, ϕ represents the neural network transformation, and σ is the activation function (Sigmoid) that produces the final prediction. The model is optimized by minimizing the Binary Cross-Entropy (BCE) loss function. Since the recommendation task was formulated using implicit feedback, observed user-item interactions were represented as positive instances (1), while unobserved interactions were treated as negative instances (0). Therefore, Binary Cross-Entropy was adopted as the optimization loss function.

$$L = -\sum_{(u,i)\in\mathcal{Y}\cup\mathcal{Y}^-} [y_{ui} \log(\hat{y}_{ui}) + (1 - y_{ui}) \log(1 - \hat{y}_{ui})] \tag{5}$$

International Journal of Emerging Research in Engineering, Science, and Management
Vol. 5 | Issue 3 | pp. 44-54 | Jul-Sep 2026.
www.ijeresm.com | eISSN – 2583-4894
Received: 14 Mar 2026 | Revised: 16 Jun 2026 | Accepted: 22 Jun 2026 | Published: 22 Jul 2026

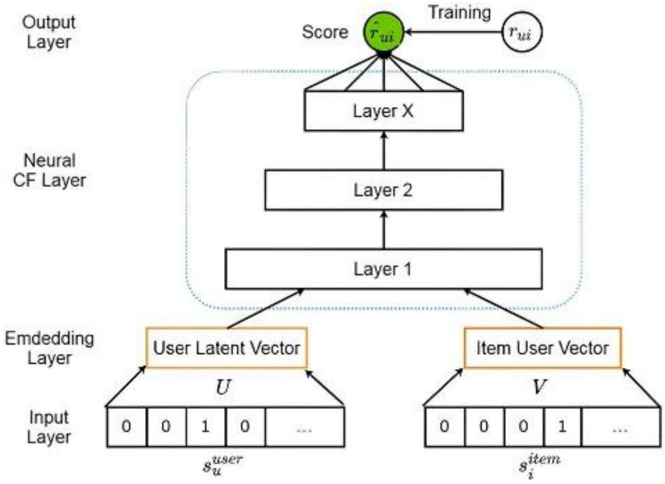

Fig. 2. NCF model formation scheme [30]

3.4. Model Evaluation

Classification performance evaluation can be done by measuring accuracy, precision, recall, and F-measure. False Positive (FP) occurs when negative data is incorrectly predicted as positive. True Positive (TP) occurs when positive data is correctly predicted as positive. True Negative (TN) occurs when negative data is predicted as negative. False Negative (FN) occurs when positive data is incorrectly predicted as negative. Precision is calculated using the equation (6).

$$Precision = \frac{TP}{TP + FP} \tag{6}$$

The equation used to calculate the recall value is shown in (7).

$$Recall = \frac{TP}{TP + FN} \tag{7}$$

F-measure is the harmonic mean of precision (P) and recall (R), which represents the system's performance. The equation used to calculate the F-measure is shown in (8).

$$F\_measure = \frac{2PR}{P + R} \tag{8}$$

3.5. Deployment Model and Interactive Dashboard Development

The final stage of this study involves deploying the recommendation system through an interactive dashboard. This dashboard was developed using Python and the Streamlit framework for its simplicity in creating responsive web applications. Initially, the dashboard was created and tested locally before being uploaded to a GitHub repository for public deployment, allowing wider accessibility [31]. The development process began with integrating essential libraries for data processing, visualization, and user interface development. Once the page configuration and external style enhancements were completed, product data was loaded efficiently to support interactive user exploration. The dashboard itself is divided into two main sections: "Dashboard" and "Application". In the Dashboard view, key performance metrics such as the number of users, available products, and total sales are presented. Data visualizations, including bar charts showing the most popular categories, subcategories, and products, were generated using the Plotly library, known for its interactive, aesthetically appealing output. For each query, the system returns the Top-10 highest-ranked product recommendations.

The Neural Collaborative Filtering (NCF) approach in the system analyzes purchasing behavior and user ratings. A user-item interaction matrix is constructed and used to learn latent user and item representations using the NCF model. Based on the learned latent representations, the system predicts user preferences and recommends items with the highest predicted relevance. This method allows the system to capture hidden patterns in user preferences. On the other hand, the Content-Based Filtering approach leverages product descriptions. These descriptions are transformed into vector representations using TF-IDF, and cosine similarity is then used to measure product similarity. As a result, when a user selects a particular product, the system can suggest other items with similar features or attributes. This helps ensure that the recommendations remain relevant even when user interaction data is limited.

International Journal of Emerging Research in Engineering, Science, and Management
Vol. 5 | Issue 3 | pp. 44-54 | Jul-Sep 2026.
www.ijeresm.com | eISSN – 2583-4894
Received: 14 Mar 2026 | Revised: 16 Jun 2026 | Accepted: 22 Jun 2026 | Published: 22 Jul 2026

4 Results and Discussions

The combination of hyperparameters, such as learning rate and batch size, is evaluated to optimize the model performance. Table 1 presents experimental results for different hyperparameter combinations, specifically the effects of learning rate and batch size on training and testing errors. The evaluated learning rates and batch sizes were selected empirically through preliminary experiments to identify a stable convergence region for the proposed model.

Table 1. Hyperparameter experiment results

Combination	Learning Rate	Batch Size	Training BCE Loss	Testing BCE Loss
Combination-1	0.01	32	2.093×10^{-3}	0.0705
Combination-2	0.01	64	1.993×10^{-3}	0.0778
Combination-3	0.01	128	6.933×10^{-3}	0.2323
Combination-4	0.01	256	1.106×10^{-3}	1.0152
Combination-5	0.100	32	3.789×10^{-3}	0.3025
Combination-6	0.100	64	1.067×10^{-3}	0.7437
Combination-7	0.100	128	8.744×10^{-3}	1.8216
Combination-8	0.100	256	8.461×10^{-3}	3.1296
Combination-9	0.001	32	3.476×10^{-3}	1.6641
Combination-10	0.001	64	1.961×10^{-3}	2.5840
Combination-11	0.001	128	3.754×10^{-3}	3.9600
Combination-12	0.001	256	1.528×10^{-3}	4.6140

Based on the results in Table 1, the combination of a learning rate of 0.01 and a batch size of 32 (Combination-1) achieves the best performance, with the lowest testing error of 0.0705, indicating an optimal balance between learning and generalization. Meanwhile, the combinations with a learning rate of 0.1 exhibited substantially higher test errors, particularly at larger batch sizes, as seen in Combination-8, which had a test error of 3.1296. On the other hand, combinations with a learning rate of 0.001 produced considerably higher testing errors, particularly Combination-12, which reached 4.6140. From this comparison, it can be concluded that the learning rate and batch size substantially influence model performance and generalization. A larger batch size is also associated with higher testing errors, possibly because it affects the model's convergence. Therefore, the combination of a learning rate of 0.01 and a batch size of 32 can be considered the optimal configuration for this model. These experimental results imply that the selection of hyperparameters significantly affects the model's performance.

The learning rate 0.01 and batch size 32 yielded the lowest average loss, so they are used to train the model to achieve better performance. The loss graph in Figure 3 shows that the model converged quickly. The training loss decreased sharply at the start of training and stabilized near zero after the 5th epoch. The model performance evaluation produced a precision value of 0.9583, a recall of 0.8251, and an F-measure of 0.8867. To further validate the effectiveness of the proposed model, a comparative evaluation was conducted against several baseline recommendation methods, including user-based and item-based collaborative filtering and matrix factorization. All baseline methods were implemented using the same dataset, preprocessing procedure, and identical 80:20 train-test split (random seed = 42) to ensure a fair comparison.

Table 2. Benchmark comparison of recommendation system performance

Model	Precision	Recall	F1-score
User-Based CF	0.81	0.72	0.76
Item-Based CF	0.84	0.75	0.79
Matrix Factorization	0.89	0.78	0.83
Proposed NCF Hybrid	0.95	0.82	0.88

As shown in Table 2, the proposed NCF-based hybrid model consistently outperforms traditional baseline methods across all evaluation metrics. This result highlights the advantage of incorporating neural-based interaction modeling in capturing complex user–item relationships, leading to more accurate recommendations. The hyperparameter tuning process revealed that the model achieves optimal convergence at 20 epochs. The learning curves show a steady decrease in Binary Cross-Entropy loss, with the test error stabilizing at 0.0705, suggesting the model learns effectively without significant overfitting. Parameter sensitivity analysis showed that a learning rate of 0.01 provided the best balance between training speed and stability; higher rates led to erratic loss fluctuations, while lower rates resulted in slower convergence. Furthermore, a batch size of 32 provided the most consistent gradient updates for this dataset.

International Journal of Emerging Research in Engineering, Science, and Management
Vol. 5 | Issue 3 | pp. 44-54 | Jul-Sep 2026.
www.ijeresm.com | eISSN – 2583-4894
Received: 14 Mar 2026 | Revised: 16 Jun 2026 | Accepted: 22 Jun 2026 | Published: 22 Jul 2026

Table 3 shows the test results of the content-based filtering method in recommending products based on similar features. For LG Smart LED TV products, the highest similarity score of 0.54 is obtained against Acer 139 Ultra HD and Acer S Series HD products. This value indicates that both products have characteristics that are quite similar to LG Smart LED TV, such as screen resolution or display technology. Other products such as the OnePlus Y Series 4K Ultra HD and Sony Bravia Ultra HD Smart LED also have similarity values above 0.40, which indicates similarity on certain features although not as high as the first two products.

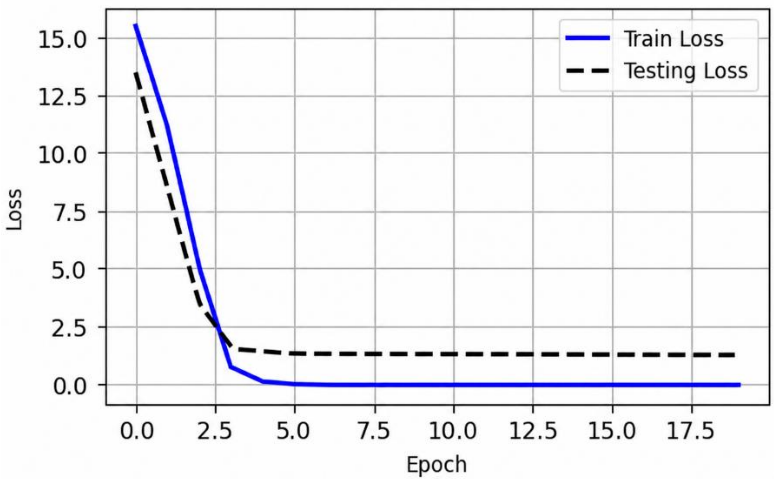

Fig. 3. Training and testing Binary Cross-Entropy loss across epochs (For Combination-1)

Meanwhile, for the Samsung Galaxy M13 (Stardust Brown), it can be seen that the other color variants of the same product, Aqua Green and Midnight Blue, obtained very high similarity scores of 0.99 and 0.98, respectively. This indicates that the system is able to accurately identify products with identical features, especially when there are only minor differences such as color. The test results of this content-based filtering-based recommendation system show an Average Precision (AP) value of 0.804 and 80% accuracy, which indicates that this system is quite effective in providing relevant product recommendations for users. A recommendation was considered correct when the cosine similarity score exceeded the predefined relevance threshold of 0.40. Average Precision (AP) was computed following the standard Information Retrieval definition using the ranked Top-K recommendations. The high AP value indicates that the system is able to provide recommendations that are relevant to user preferences, while the accuracy rate of 80% indicates that the majority of recommendations generated are indeed in accordance with the target product.

To validate the effectiveness of the proposed hybrid NCF-CBF model, a comparison was conducted against standalone baseline algorithms. Experimental results show that while standalone Content-Based Filtering achieved an accuracy of 80% and traditional KNN-based Collaborative Filtering struggled under the high dataset sparsity (99.91%), the proposed hybrid model demonstrated superior performance. The sparsity level was calculated as $1 - \frac{1465}{1191*1334}$, where 1465 represents the observed user-item interactions, while 1191 and 1334 denote the numbers of unique users and products, respectively. The integration of neural layers allowed for capturing non-linear interactions that are often missed by Matrix Factorization, resulting in a significantly lower testing error (0.0705) and higher precision (0.9583). This comparison demonstrates that the hybrid approach effectively mitigates the limitations of individual baseline models in a sparse e-commerce environment.

Table 3. Sample content-based filtering test results

Product Name			
LG Smart LED TV		Samsung Galaxy M13 (Stardust Brown)	
Product Similarity	**Similarity Score**	**Product Similarity**	**Similarity Score**
Acer 139 Ultra HD	0.54	Samsung Galaxy M13 (Aqua Green)	0.99
Acer S Series HD	0.54	Samsung Galaxy M13 (Midnight Blue	0.98
OnePlus Y Series 4K Ultra HD	0.47	Samsung Galaxy M13 (Stardust Brown)	0.99
Sony Bravia Ultra HD Smart LED	0.45	-	-

The e-commerce recommendation system dashboard, as shown in Fig. 4 and Fig. 5, is designed to provide a more personalized and relevant user experience. By leveraging transaction data and user preferences, the system can improve customer satisfaction by presenting product recommendations that align with their needs and interests. The dashboard displays important information, such as the number of users, available products, and total sales.

International Journal of Emerging Research in Engineering, Science, and Management
Vol. 5 | Issue 3 | pp. 44-54 | Jul-Sep 2026.
www.ijeresm.com | eISSN – 2583-4894
Received: 14 Mar 2026 | Revised: 16 Jun 2026 | Accepted: 22 Jun 2026 | Published: 22 Jul 2026

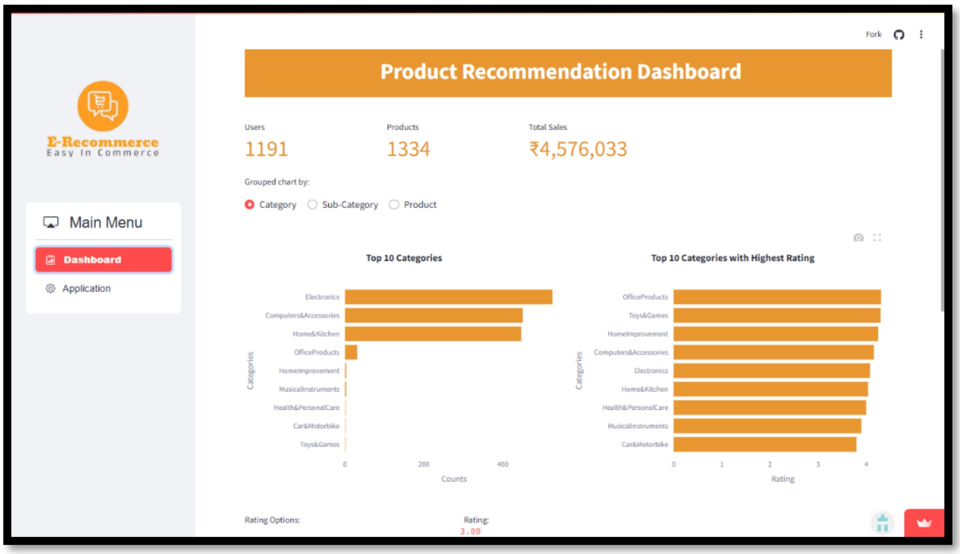

Fig. 4. Dashboard menu of e-commerce recommendation system

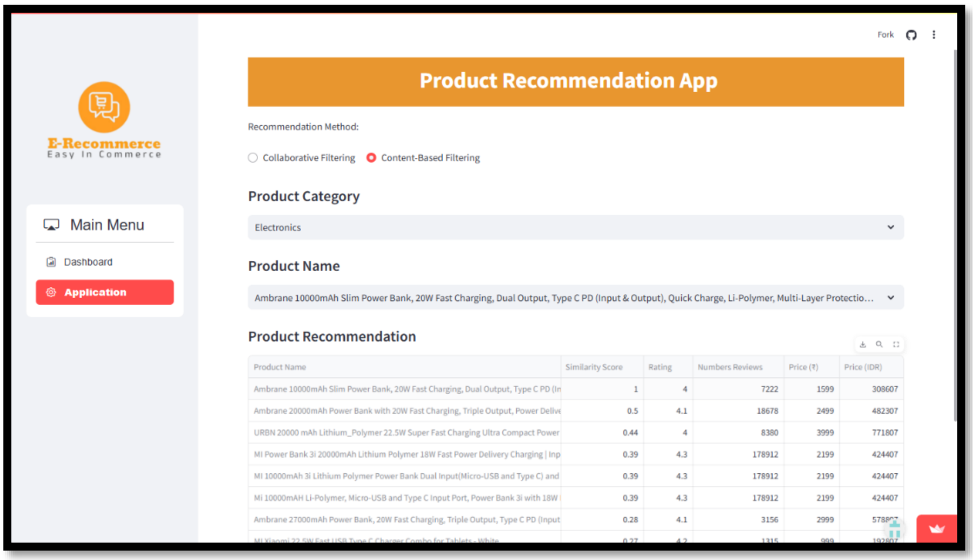

Fig. 5. Application menu of e-commerce recommendation system

In addition, there are visual charts that display the most popular product categories and the highest-rated categories, making it easier for users to understand product trends. The e-commerce recommendation system dashboard can be accessed through the following link: https://ecommerce-recommendation-system.streamlit.app/. This recommendation system works using Collaborative Filtering and Content-Based Filtering. Neural Collaborative Filtering predicts user preferences by learning latent user-item interactions through neural network embeddings. Meanwhile, Content-Based Filtering recommends products based on similar characteristics of products users have purchased or searched for. Users can select the appropriate recommendation method through the application menu, as shown in Fig. 5.

To facilitate product search, users can select a particular category. The system will then display a list of recommended products based on similarity score, rating, number of reviews, and price. With this information, users can compare products more easily before purchasing. Overall, the e-commerce recommendation system dashboard improves users' shopping experience by enabling more accurate, relevant product discovery. By integrating innovative recommendation methods, the system enables users to efficiently find products that best suit their preferences. The scope of this study is restricted to a single Amazon dataset and a functional system evaluation. Subsequent research may include multiple datasets, integration of real-time feedback, and user-based assessments.

5 CONCLUSION

This study proposes a hybrid recommendation system integrating Neural Collaborative Filtering and content-based filtering within an interactive dashboard framework. The results demonstrate that the proposed approach not only improves recommendation accuracy but also enhances interpretability through interactive visualization. The main contribution of this study lies in combining advanced neural modeling with user-centric system design, supporting the integration of predictive performance and practical usability in e-commerce environments. The current implementation was evaluated only through functional and computational testing, and no formal user-based usability study was conducted. The present study is limited to a single Amazon dataset and functional system evaluation. Future work may validate the proposed framework using multiple datasets, incorporate contextual features and real-time user feedback, and conduct comprehensive user-based evaluations to further improve recommendation performance.

ACKNOWLEDGEMENTS

The authors would like to thank Amazon for providing access to the Amazon Sales Dataset used in this study.

FUNDING INFORMATION

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

GENERATIVE AI DECLARATION

During the preparation of this manuscript, the authors used ChatGPT and Grammarly exclusively to assist with proofreading, editing, and formatting. These AI tools were not used to generate, analyze, interpret, or fabricate scientific data or research findings. All content was subsequently reviewed, revised, and approved by the authors, who assume full responsibility for the accuracy, integrity, and originality of the published work.

ETHICS STATEMENT

This study did not involve human or animal subjects and, therefore, did not require ethical approval.

STATEMENT OF CONFLICT OF INTERESTS

The authors declare no conflicts of interest related to this study.

LICENSING

REFERENCES

[1] M. A. Nazir and M. A. Roomi, "Barriers to adopting electronic commerce for small and medium-sized enterprises in emerging economies," *EMAJ Emerging Markets Journal*, vol. 10, no. 2, pp. 43–55, Apr. 2021, doi: 10.5195/emaj.2020.203.

[2] Statista, "Number of users of e-commerce in Indonesia 2017-2029," *Statista*, Feb. 03, 2026. https://www.statista.com/forecasts/251635/e-commerce-users-in-indonesia/ ((accessed Feb. 5, 2026).

[3] X. He, Q. Liu, and S. Jung, "The Impact of recommendation system on User satisfaction: A moderated mediation approach," *Journal of Theoretical and Applied Electronic Commerce Research*, vol. 19, no. 1, pp. 448–466, Feb. 2024, doi: 10.3390/jtaer19010024.

[4] N. Ghanem, S. Leitner, and D. Jannach, "Balancing consumer and business value of recommender systems: A simulation-based analysis," *Electronic Commerce Research and Applications*, vol. 55, p. 101195, Aug. 2022, doi: 10.1016/j.elerap.2022.101195.

[5] A. Sami, W. E. Adrousy, S. Sarhan, and S. Elmougy, "A deep learning based hybrid recommendation model for internet users," *Scientific Reports*, vol. 14, no. 1, p. 29390, Nov. 2024, doi: 10.1038/s41598-024-79011-z.

[6] Y. Yun, D. Hooshyar, J. Jo, and H. Lim, "Developing a hybrid collaborative filtering recommendation system with opinion mining on purchase review," *Journal of Information Science*, vol. 44, no. 3, pp. 331–344, Feb. 2017, doi: 10.1177/0165551517692955.

[7] H. Morise, K. Atarashi, S. Oyama, and M. Kurihara, "Neural collaborative filtering with multicriteria evaluation data," *Applied Soft Computing*, vol. 119, p. 108548, Feb. 2022, doi: 10.1016/j.asoc.2022.108548.

[8] M. F. Aljunid and M. Dh, "An efficient deep learning approach for collaborative filtering recommender system," *Procedia Computer Science*, vol. 171, pp. 829–836, Jan. 2020, doi: 10.1016/j.procs.2020.04.090.

[9] G. Tschinkel, "A recommendation dashboard utilising micro-filter visualisations: An evaluation and investigation of applicability on mobile devices," *Graphical Models*, vol. 110, p. 101074, May 2020, doi: 10.1016/j.gmod.2020.101074.

[10] S. Jayalakshmi, N. Ganesh, R. Čep, and J. S. Murugan, "Movie Recommender Systems: Concepts, methods, challenges, and future directions," *Sensors*, vol. 22, no. 13, p. 4904, Jun. 2022, doi: 10.3390/s22134904.

[11] L. Zangari, S. Raso, and A. Tagarelli, "Transformers for graph-based Recommender Systems: A survey," *Machine Intelligence Research*, vol. 23, no. 1, pp. 34–69, Feb. 2026, doi: 10.1007/s11633-025-1607-8.

[12] F. O. Isinkaye, Y. O. Folajimi, and B. A. Ojokoh, "Recommendation systems: Principles, methods and evaluation," *Egyptian Informatics Journal*, vol. 16, no. 3, pp. 261–273, Aug. 2015, doi: 10.1016/j.eij.2015.06.005.

[13] C. Channarong, C. Paosirikul, S. Maneeroj and A. Takasu, "HybridBERT4Rec: A Hybrid (Content-Based Filtering and Collaborative Filtering) Recommender System Based on BERT," in *IEEE Access*, vol. 10, pp. 56193-56206, 2022, doi: 10.1109/ACCESS.2022.3177610.

[14] Q. Li and J. Kim, "A Deep Learning-Based Course Recommender System for Sustainable Development in education," *Applied Sciences*, vol. 11, no. 19, p. 8993, Sep. 2021, doi: 10.3390/app11198993.

[15] J. Kunkel and J. Ziegler, "A comparative study of item space visualizations for recommender systems," *International Journal of Human-Computer Studies*, vol. 172, p. 102987, Dec. 2022, doi: 10.1016/j.ijhcs.2022.102987.

[16] Y. Akkem, B. S. Kumar, and A. Varanasi, "Streamlit Application for advanced ensemble learning methods in CrOP Recommendation Systems – a review and implementation," *Indian Journal of Science and Technology*, vol. 16, no. 48, pp. 4688–4702, Dec. 2023, doi: 10.17485/ijst/v16i48.2850.

[17] M. Saxborn, Y. Pan, and A. Said, "Trust Through Recommendation in E-commerce," in *Proceedings of the 2024 ACM SIGIR Conference on Human Information Interaction and Retrieval*, New York, NY, USA: ACM, Mar. 2024, pp. 87–96. doi: 10.1145/3627508.3638294.

[18] S. Shahidi, A. W. Samadzai, and H. Shahbazi, "Effective data preprocessing in data Science: From method selection to Domain-Specific Optimization," *Journal of Advanced Computer Knowledge and Algorithms*, vol. 2, no. 4, pp. 84–90, Jul. 2025, doi: 10.29103/jacka.v2i4.22886.

[19] R. J. Little, J. R. Carpenter, and K. J. Lee, "A comparison of three popular methods for handling missing data: Complete-Case analysis, inverse probability weighting, and multiple imputation," *Sociological Methods & Research*, vol. 53, no. 3, pp. 1105–1135, Aug. 2022, doi: 10.1177/00491241221113873.

[20] J. Deng, H. Li, J. Guo, L. Y. Zhang, and Y. Wang, "Providing prediction reliability through deep neural networks for recommender systems," *Computers & Industrial Engineering*, vol. 185, p. 109627, Sep. 2023, doi: 10.1016/j.cie.2023.109627.

[21] E. Nguyen, "Text mining and network analysis of digital libraries in R," in *Elsevier eBooks*, 2013, pp. 95–115. doi: 10.1016/b978-0-12-411511-8.00004-9.

[22] S. N. Putri, T. A. Zuraiyah, and D. M. Akhmad, "Recommender Systems using Hybrid Demographic and Content-Based Filtering methods for UMKM Products," *Komputasi Jurnal Ilmiah Ilmu Komputer Dan Matematika*, vol. 21, no. 1, pp. 31–44, Jan. 2024, doi: 10.33751/komputasi.v21i1.8991.

[23] S. Zhu, J. Wu, H. Xiong, and G. Xia, "Scaling up top-K cosine similarity search," *Data & Knowledge Engineering*, vol. 70, no. 1, pp. 60–83, Sep. 2010, doi: 10.1016/j.datak.2010.08.004.

[24] P. Ortal and M. Edahiro, "Similarity Measure for Product Attribute Estimation," in *IEEE Access*, vol. 8, pp. 179073-179082, 2020, doi: 10.1109/ACCESS.2020.3027023.

[25] B. Wu and J. Qin, "A List-Ranking Framework Based on Linear and Non-Linear Fusion for Recommendation from Implicit Feedback," *Entropy*, vol. 24, no. 6, p. 778, May 2022, doi: 10.3390/e24060778.

[26] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, "Neural Collaborative Filtering," in *Proceedings of the 26th International Conference on World Wide Web*, Republic and Canton of Geneva, Switzerland: International World Wide Web Conferences Steering Committee, Apr. 2017, pp. 173–182. doi: 10.1145/3038912.3052569.

[27] W. Chen, F. Cai, H. Chen, and M. De Rijke, "Joint neural collaborative filtering for recommender systems," *ACM Transactions on Information Systems*, vol. 37, no. 4, pp. 1–30, Aug. 2019, doi: 10.1145/3343117.

[28] S. R. Dubey, S. K. Singh, and B. B. Chaudhuri, "Activation functions in deep learning: A comprehensive survey and benchmark," *Neurocomputing*, vol. 503, pp. 92–108, Jul. 2022, doi: 10.1016/j.neucom.2022.06.111.

[29] H.-H. Chen, C.-A. Chung, H.-C. Huang, and W. Tsui, "Common pitfalls in training and evaluating recommender systems," *ACM SIGKDD Explorations Newsletter*, vol. 19, no. 1, pp. 37–45, Sep. 2017, doi: 10.1145/3137597.3137601.

[30] S. Feng, K. Song, D. Wang, W. Gao, and Y. Zhang, "InterSentiment: combining deep neural models on interaction and sentiment for review rating prediction," *International Journal of Machine Learning and Cybernetics*, vol. 12, no. 2, pp. 477–488, Aug. 2020, doi: 10.1007/s13042-020-01181-9.

[31] K. Verbert, P. Brusilovsky, C. Wongchokprasitti, D. Parra, and B. Cardoso, "Supporting Conference Attendees with Visual Decision Making Interfaces," in *Companion proceedings of the 22nd international conference on intelligent user interfaces*, New York, NY, USA: ACM, Mar. 2017, pp. 161–164. doi: 10.1145/3030024.3038273.