Personalized Product Recommendations Using Restricted Boltzmann Machines To Overcome Cold-Start Challenges On A Niche Coffee E-Commerce Platform

Emilia Hesti [a,1], Ade Silvia Handayani [a,2,*], Suzanzefi [a,3], Muhammad Zakuan Agung [a,4], Ella Rosita [b,5], Asriyadi [c,6], Afifah Syifah Kaila [a,7], Luthfia Afifah [a,8], M. Ardiansyah [a,9]

[a] Department of Electrical Engineering, Applied Bachelor Program in Telecommunication Engineering, State Polytechnic of Sriwijaya, Palembang, Indonesia
[b] Palembang Training Center and Sriwijaya Coffee Community, Palembang, Indonesia
[c] Electronics And Communication Engineering Department, King Abdul Aziz University, Jeddah, Saudi Arabia
[1] emiliahesti@ymail.com; [2] ade_silvia@polsri.ac.id*; [3] Suzanzefi250977@gmail.com; [4] mzakuanagung30115@gmail.com; [5] ellarosita@yahoo.com; [6] aasriyadi@stu.kau.edu.sa; [7] afifahsyifahk@gmail.com; [8] luthfiaafifah58@gmail.com; [9] mhdarrdnsy@gmail.com
* corresponding author

ARTICLE INFO

Article history
Received
Revised
Accepted

Keywords
Cold-start
E-commerce
Recommendation System
Restricted Boltzmann Machine

ABSTRACT

This paper examines the use of a Restricted Boltzmann Machine (RBM) to provide personalized product recommendations on a niche coffee e-commerce platform facing cold-start conditions. We train RBM variants on a binary transaction matrix derived from 100 simulated user transactions and evaluate four hidden-unit configurations (3, 5, 10, 15) using 5-fold cross-validation. Models were trained with Contrastive Divergence (CD-1) and assessed primarily by Mean Squared Error (MSE) for reconstruction fidelity, complemented by ranking metrics (Precision@3, NDCG@3). The 10-hidden-unit configuration achieved the best balance of reconstruction and ranking performance, with an average test MSE ≈ 0.0454, outperforming popular-item (MSE: 0.0802) and random (MSE: 0.0760) baselines. While the RBM demonstrates strong capability in modeling latent user preferences under sparse data, ranking metrics expose limitations when predicting exact top-N items in extremely sparse cases. The study highlights practical implications for early-stage niche marketplaces and suggests integrating content signals or hybridization to further improve top-N recommendation quality.

1. Introduction

Information technology has experienced rapid advancement. This progress significantly drives digital transformation across various sectors. The commercial domain is one notable example. E-commerce represents a popular innovation among consumers. Its easy access and flexible shopping experience contribute to its popularity [1]. For niche e-commerce platforms, such as those focused on coffee products, the ability to deliver a personalized experience is a key differentiator. However, such platforms often face a critical obstacle, namely the ‘cold-start problem,’ where the absence of historical interaction data prevents recommendation systems from functioning effectively [2]. Integrating artificial intelligence (AI)-based recommendation systems emerged as a promising solution to this issue. Numerous approaches have been utilized in the development of recommendation systems, including content-based filtering, collaborative filtering, and model-based techniques such as neural networks. The cold-start problem has three primary manifestations. The "new user" problem occurs when customer preference data is unavailable. The "new item" problem arises when a product has insufficient interaction data for recommendations. The "new system" problem affects new platforms with a general lack of initial data [3]. For a niche e-commerce platform like the one in this study, this challenge is particularly detrimental. A new user who receives generic or irrelevant coffee recommendations may quickly lose interest, leading to lost sales opportunities and a poor initial user experience that hinders long-term customer retention.

Several techniques exist to mitigate these challenges, with content-based filtering being a prominent approach. This method functions by recommending items based on an analysis of their attributes. It demonstrates particular effectiveness in addressing the new item cold-start problem, as a new coffee product can be immediately recommended based on its metadata (e.g., bean origin, roast level, flavor notes) without any prior purchase history. However, content-based systems are heavily reliant on rich and well structured metadata. They can lead to over-specialization, often failing to recommend novel items outside a user's established taste profile . Consequently, many modern systems employ hybrid models that combine collaborative and content-based techniques to leverage the strengths of both [4].

To address the cold-start challenge, various techniques have been proposed, ranging from traditional collaborative filtering approaches [7] to more advanced deep learning architectures [13]. While traditional methods can be effective, they often struggle with the extreme data sparsity inherent in cold-start scenarios. Deep learning models, including Deep Belief Networks (DBN) [30], have shown promise in capturing more complex user-item relationships. Among these, the Restricted Boltzmann Machine (RBM), a generative stochastic neural network, has emerged as a powerful unsupervised method for collaborative filtering [5].

Previous research has demonstrated the effectiveness of RBM in various recommendation domains. For instance, Hazrati and Elahi [5] incorporated visual features with RBM to address the new item problem in video recommendations. Similarly, hybrid models combining RBM with other techniques like K-Nearest Neighbors have been successfully applied for movie recommendations [16]. Other studies have also explored variations of RBM for different recommendation tasks [15, 18, 28]. However, much of the existing research focuses on platforms with established data or specific types of cold-start (e.g., new item). The application and performance evaluation of RBM in the unique context of a newly launched, niche e-commerce platform simulating a "system cold-start" where almost no data exists, remains an underexplored area. Therefore, the novelty of this research lies in its practical implementation and analysis of RBM's capability to model user preferences under conditions of extreme data sparsity, providing a specific case study for SME e-commerce platforms.

Among these various approaches, the Restricted Boltzmann Machine (RBM) has presented a great potential as an unsupervised learning mode in building effective recommendation systems, especially in scenarios with limited data, as in the case of the cold-start problem [5].

While these methods have their merits, this study focuses on the Restricted Boltzmann Machine (RBM) due to its unique advantages in the context of a new and data-scarce platform. Unlike many traditional models, RBMs excel at learning from implicit feedback in this case, purchase data alone, without requiring explicit user ratings. More importantly, as a generative model the RBM possesses the capability to learn deep, latent features. These learned features are crucial because they represent non-obvious relationships that exist between different products. This allows the model's understanding to extend beyond simple popularity patterns and grasp the underlying tastes of users, a function that is vital when historical data is scarce [5].

Characterized as a probabilistic graphical model, the Restricted Boltzmann Machine (RBM) is a specific type of stochastic neural network. An energy-based approach is central to the model’s core learning process. This is accomplished through an energy function that assigns a unique scalar value for every possible configuration between the visible and hidden units. The model's objective is to adjust its weights and biases to learn a probability distribution over the data that minimizes this energy function for observed data points. In simpler terms, more plausible configurations representing common user purchase patterns are assigned a lower energy and thus have a higher probability of being generated by the model. This energy-based approach enables the RBM to capture complex, nonlinear relationships within the data, making it a powerful tool for collaborative filtering [5]. Through their preliminary research, Hinton and Salakhutdinov have demonstrated the significant effectiveness of RBM for collaborative filtering tasks. Unlike conventional matrix factorization methods, RBM learns from latent relationships between products, not just interaction frequency. Several prior studies have confirmed that RBMs, as energy-based models, can generate more personalized recommendations compared to traditional techniques particularly in sparse data environments such as e-commerce platforms [6].

This capability is relevant because one of the most significant challenges in recommendation systems is the cold-start problem, which occurs when insufficient data is available for new users or products. RBM has been successfully applied to mitigate this limitation. However, the application of RBM in particular e-commerce domains, such as coffee products, especially for newly launched platforms with limited data, has been rarely explored. This research attempts to fill this gap and aims to implement the RBM algorithm on a newly developed coffee e-commerce platform, evaluating its performance through Mean Squared Error (MSE) and training time. RBM's recommendations are compared to baseline methods of popular item and random recommendation to determine their relative performance [7].

This study is expected to contribute to the development of AI-based recommendation systems in the e-commerce sector, particularly for platforms that operate with limited historical data. The application of RBM in the context of coffee product sales offers a specific and practical case study that enriches the body of literature on recommendation systems for small and medium enterprises (SMEs) and local product markets.

2. Method

In this research, the Restricted Boltzmann Machine (RBM) algorithm serves as the main component for developing the recommendation model through an experimental-based implementation approach. All processes from implementation and training to evaluation were carried out using Python and the PyTorch framework on Google Colab, enabling efficient data processing and visualization of output. Fig 1 illustrates the workflow of the proposed recommendation system, which begins with user data input, proceeds with model training, and culminates in product recommendations as output.

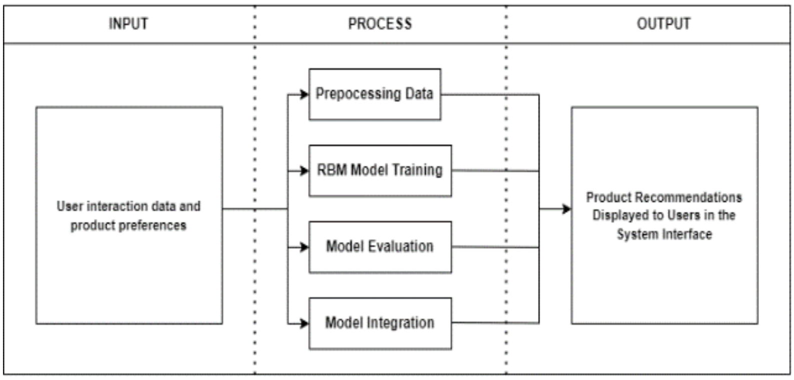

Fig. 1. Workflow diagram of the RBM-based product recommendation system

The workflow for implementing the system is illustrated in Fig. 1. The process begins with the Input stage, which consists of user interaction data. This is followed by the Process stage, which includes data preprocessing, RBM model training, evaluation, and model integration, ultimately producing Output in the form of product recommendations for users

2.1. Dataset and Preprocessing

The experiment uses a simulated transaction dataset representing a newly-launched coffee marketplace. The dataset contains 100 user transactions and 48 unique products; the binary user–item matrix entries indicate purchase presence (1) or absence (0). The dataset was intentionally designed to be highly sparse (≈95%) to emulate cold-start conditions. We applied standard preprocessing steps (deduplication, binarization) and used 5-fold cross-validation to obtain stable estimates of model performance. In this process, the dataset was partitioned into five distinct subsets, commonly known as "folds." The model underwent iterative training on four folds while being tested on the one remaining subset. This procedure was repeated five times to ensure that every subset served once as the test set. The final performance metric for each RBM configuration was calculated by averaging the Mean Squared Error (MSE) across all five test iterations. This approach minimizes the risk of performance bias that can arise from a single, arbitrary train-test split. It provides a more stable estimate of the model's effectiveness on unseen data [8]. It is important to clarify that this study utilizes a simulated dataset instead of collecting real-world data or using questionnaires. This methodological choice is deliberate and essential for addressing the core research problem: the "system cold-start" scenario. In such a scenario, where an e-commerce platform is newly launched, historical transaction data is inherently unavailable. Therefore, simulating a highly sparse dataset is the most effective approach to replicate this specific condition and evaluate the model's performance from day one.

This study uses a simulation dataset to reflect coffee product transactions on a new e-commerce platform. This dataset was deliberately created with a sparsity level of 95% to mimic cold-start conditions, where most products have minimal purchase history. A sample of 100 user transactions was taken from this dataset to form the experimental data. This data was then transformed into a binary user-item matrix with 100 rows (representing unique user transactions) and 48 columns (representing unique products). The detailed specifications of the data used are summarized in Table 1.

Table 1. Specification of Simulated Data

Component	Value
Number of Products	48
Number of Users	100
Sparsity Level	95%
Data Format	Binary Matrix (0/1)
Data Source	Simulated Coffee Transaction

2.2. RBM Model Architecture and Training

We implemented a binary-visible binary-hidden RBM trained with Contrastive Divergence (CD-1). To improve reproducibility, record the initialization and training settings explicitly in the manuscript (e.g., weight initialization method, optimizer details, random seed, batch size). In our experiments we compared four hidden-unit sizes (3, 5, 10, 15) using the following training configuration: 50 epochs, learning rate 0.05, and MSE loss for reconstruction. For clarity, we also provide the high-level training loop in pseudo-code (appendix or supplementary material) so that the procedural description does not rely solely on textbook definitions.

- Epochs: 50
- Learning rate: 0.05
- Loss function: Mean Squared Error (MSE)
- Training Algorithm: Contrastive Divergence (CD-1)

An experimental workflow was established in which the RBM algorithm served as the core engine for personalized recommendations on a simulated e-commerce dataset as the core of a recommendation system for e-commerce platforms. In this implementation, RBM is modeled as a stochastic two-layer network, where hidden neurons capture latent representations of coffee product preferences through cross-layer interactions, in which the visible and hidden layers interact through cross-layer connections while remaining independent within each layer. The model is designed to learn latent patterns in users' purchase behavior. These training parameters were established based on preliminary experiments to ensure effective learning. The learning rate of 0.05 was selected as it provided stable convergence during training, offering a good balance between model accuracy and computational efficiency. Similarly, 50 epochs was determined to be sufficient for the model's reconstruction error to converge on the training data without exhibiting significant signs of overfitting on the test sets used in the cross-validation process.

2.3. Evaluation Metric

The input data for this model is structured as a binary transaction matrix. In this matrix, each row represents a unique order, while each column represents a specific product. A value of 1 indicates the presence of a product in an order, and a value of 0 indicates its absence. The training process consisted of 50 epochs, during which weights were updated using the Contrastive Divergence algorithm [9]. Model performance was evaluated by computing the Mean Squared Error (MSE) between the original input and the reconstructed output on the test data. Training time was recorded for each configuration to assess learning efficiency. A quantitative evaluation metric is needed to measure the model's accuracy and reliability. This study chose Mean Squared Error (MSE) as the primary metric to assess the model's performance [10]. The evaluation includes measuring prediction accuracy using MSE and analyzing training efficiency across different configurations of hidden units. The Mean Squared Error (MSE) metric quantifies how accurately the RBM reconstructs unseen transactions, serving as a direct indicator of model generalization in Equation (1):

MSE (Mean Squared Error):

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (v_i - \hat{v}_i)^2 \tag{1}$$

Where v_i is the actual value, $\hat{v}_i$ is the reconstructed value predicted by the RBM, and n n is the total number of input elements.

In addition to MSE for reconstruction accuracy, This metric quantifies the reconstruction fidelity of the RBM; lower values indicate closer reconstructions. Precision@k / Recall@k / NDCG@k:measures the proportion of relevant items found in the top-k recommendations [11]. They are defined as:

Precision@k:

$$\text{Precision@k} = \frac{|Recommended_k \cap\ Relevant|}{k} \tag{2}$$

Recall@k:

$$\text{Recall@k} = \frac{|Recommended_k \cap\ Relevant|}{|Relevant|} \tag{3}$$

Where k is set to 3 for this study. "Relevant Items" are the products a user actually purchased in the test set, and "Recommended Items @k" are the top-k products suggested by the model.

To evaluate the quality of the ranking position, Normalized Discounted Cumulative Gain (NDCG@k) was also used. NDCG assesses whether relevant items are placed at the top of the recommendation list [12]. It is calculated by dividing the Discounted Cumulative Gain (DCG) by the Ideal Discounted Cumulative Gain (IDCG), as shown in Equation (4).

$$NDCG@K = \frac{DCG@K}{IDCG@K} \tag{4}$$

Where DCG@k is defined in Equation (5):

$$DCG@K = \sum_{i=1}^{K} \frac{\text{rel}i}{log_2\ (i\ +\ 1)} \tag{5}$$

In Equation (5), rel_i is the relevance of the item at position $_i$ (1 if relevant, 0 otherwise). IDCG@k is the DCG score of a perfectly ranked list. This metric rewards models for placing more relevant items higher in the recommendation list.

2.4. Comparative Method (Baseline)

This study implemented two baseline methods, namely popular item recommendations and random recommendations. The popular item method is implemented by recommending the top 3 products with the highest total purchase frequency across the entire dataset. Meanwhile, the random recommendation method is implemented by selecting three products at random from the whole list of available products, ensuring no duplicates appear in a single set of recommendations. This comparison was conducted to evaluate the superiority of the RBM approach in providing more relevant and personalized recommendations [8][13].

The researchers compared the output of the RBM model with two conventional comparison models to establish a performance benchmark: a popular item recommender and a random recommender. The popular item method ranked products based on overall purchase frequency, whereas the random recommendation selected products without considering user preferences. These benchmarks were used to assess the effectiveness of the RBM in generating more relevant and personalized suggestions [13]

3. Results and Discussion

In this chapter, the study presents objective research findings based on the conducted experiments. The presentation includes the optimization results of the model architecture, the performance comparison with baseline methods, and the specific analysis of the model's capability in handling cold-start scenarios. Each result is described clearly to emphasize the effectiveness and reliability of the proposed model in practical applications.

3.1. Model Architecture Optimization via Cross-Validation

The RBM model was trained and evaluated across four distinct hidden unit configurations (3, 5, 10, and 15) to identify the optimal architecture. Performance was measured using 5-fold cross-validation to ensure the stability and reliability of the results. Table 2 summarizes the average Mean Squared Error (MSE) and standard ranking metrics for each configuration on the test data.

Table 2. Performance Evaluation of RBM with Different Hidden Unit Configurations

Hidden Units	Avg. MSE (Test Data)	Avg. Precision@3	Avg. Recall@3	Avg. NDCG@3
3	0.0477	0.035	0.090	0.013
5	0.0460	0.041	0.110	0.016
10	0.0454	0.050	0.150	0.020
15	0.0483	0.037	0.100	0.014

Table 2 reports the cross-validated performance for each hidden-unit configuration. Increasing the hidden layer size from 3 to 10 reduced average test MSE, suggesting improved capacity to capture latent purchase patterns; however, the 15-unit model shows a slight increase in MSE indicative of overfitting on this small dataset. Ranking metrics (Precision@3, Recall@3, NDCG@3) also peak at the 10-unit setting, supporting its selection as the operational model. These results imply that for highly sparse, niche datasets a moderate hidden dimensionality balances expressiveness and generalization.

The RBM's architecture consists of two layers: a visible layer, which represents observed variables (such as coffee products), and a hidden layer, which captures the latent user preferences. The model was trained using the Contrastive Divergence (CD-1) algorithm to minimize the gap between input data and its reconstruction.

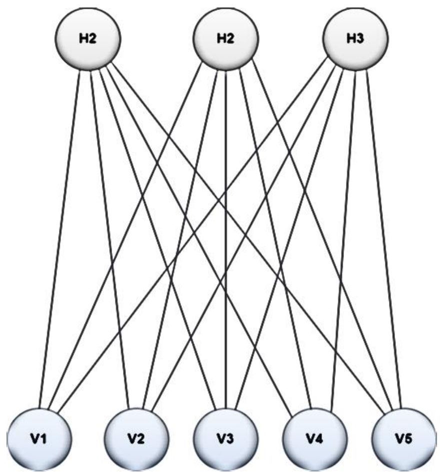

Fig. 2. Illustration of the Restricted Boltzmann Machine (RBM) architecture with five visible units and three hidden units

The recommendations generated by the RBM model were compared against two baseline methods: popular items and random recommendations. This figure illustrates the fundamental structure of the RBM used in this study. The bottom layer consists of visible units (V1-V5), which represent the observable data—in this case, the individual coffee products in the transaction matrix. The top layer contains hidden units (H1-H3), which learn to capture latent features or underlying patterns in user preferences that are not explicitly visible in the data. The key characteristic of the "Restricted" architecture is that connections only exist between the visible and hidden layers, with no connections within a layer itself. This structure allows for efficient training using the Contrastive Divergence algorithm.

3.2. Comparative Performance Against Baseline Methods

To contextualize the RBM's effectiveness, its performance was benchmarked against two standard baseline methods: popular items and random recommendations. The RBM model's optimal architecture (10 hidden units) demonstrated markedly superior performance in data reconstruction.

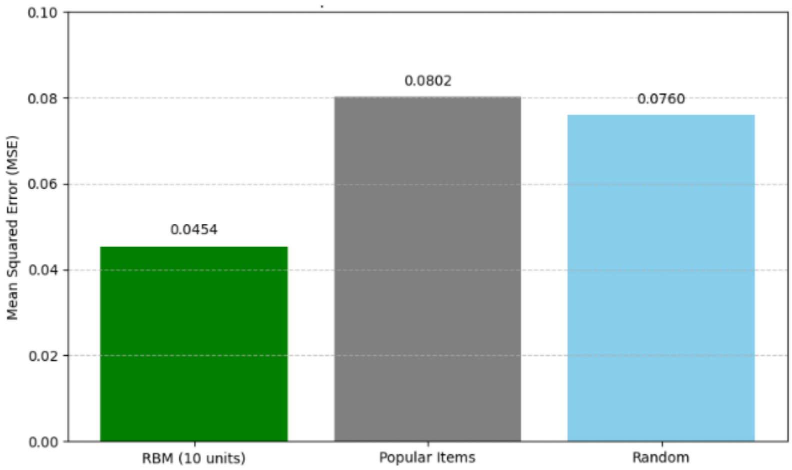

Fig. 3. Comparison of Mean Squared Error (MSE) between RBM and Baseline Methods

The Comparison of Mean Squared Error (MSE) between RBM and Baseline Methods. The MSE value for the RBM model (10 units) is the average result of 5-fold cross-validation (MSE = 0.0454). In contrast, the values for the baseline methods (Popular Items and Random) are calculated from evaluations on a single representative data set (holdout) to determine their baseline performance.

The visual comparison of the MSE values is presented in Fig 3, this significant performance gap highlights the fundamental limitation of non-personalized models. While the popular item method captures general trends, it fails to cater to niche tastes. The RBM, on the other hand, successfully models these individual user preferences by learning the underlying relationships between products, resulting in a more accurate preference reconstruction.

3.3. Analysis of Recommendation Ranking Performance

A special analysis was conducted on a subset of 'extreme cold-start' users to measure the model's robustness in the most minimal data scenario. This subset consists of users with only one transaction interaction in the dataset. Performance in this user group was evaluated using MSE and NDCG@3 metrics to understand the direct impact of extremely high data sparsity on the accuracy and quality of the recommendations generated by the model. While MSE measures the model's ability to reconstruct user preferences, metrics such as Precision, Recall, and Normalized Discounted Cumulative Gain (NDCG) evaluate the quality of the top-ranked recommendations. For the optimal 10-unit model, the ranking metrics were calculated as Precision@3 : 0.05, Recall@3 : 0.15, and NDCG@3 : 0.02.

These relatively low scores are not an indication of model failure but rather a testament to the acute difficulty of the cold-start problem. They reveal a crucial insight: the RBM excels at understanding a user's overall taste profile (low MSE). However, accurately predicting the exact top 3 items for a user with minimal data remains a significant challenge. This distinction between reconstruction accuracy and ranking accuracy is fundamental in data-scarce environments.

Table 3. Epoch-wise Reduction of Train and Test MSE for RBM with 10 Hidden Units

Epoch	Train MSE	Test MSE
1	0.0912	0.0783
10	0.0674	0.0571
25	0.0520	0.0487
50	0.0451	0.0454

Table 3 demonstrates how the RBM's reconstruction error steadily converges within 50 epochs, with a consistent decrease in both training and test MSE. This highlights the model's stability and learning effectiveness across epochs.

This table demonstrates the learning progression of the optimal RBM model (with 10 hidden units) over 50 epochs. It shows a consistent decrease in both the training MSE and the test MSE, indicating that the model effectively learned from the data without significant overfitting. By the final epoch, the test MSE (0.0454) closely converges with the train MSE (0.0451), highlighting the model's stability and its ability to generalize well to unseen data. This validates that 50 epochs were sufficient for achieving effective model training.

3.4. Cold-Start Performance Analysis

As confirmed in an extreme cold-start test (users with one interaction), ranking precision dropped to 0.005, validating the difficulty of sparse data conditions discussed in Section 3.3

Table 4. Performance Comparison Between General Users and Cold-Start Users

Method	Avg. MSE	Precision@3	NDCG@3	User Type
RBM (10 units)	0.0454	0.050	0.020	All Users
RBM (10 units)	0.0437	0.015	0.0053	Cold-Start (1 interaction)

As shown in Table 4, the model demonstrates consistent reconstruction performance across user types, with nearly identical MSE scores. However, the ranking quality, as reflected by Precision@3 and NDCG@3, drops significantly in the cold-start group. This supports the assertion that while RBM effectively captures latent structure, its top-k recommendation precision is constrained under data-scarce conditions. The findings reinforce the importance of hybrid or content-enhanced systems for addressing the cold-start problem.

These results highlight the RBM's strength in reconstructing general preferences while also demonstrating its limitations in top-N recommendation performance under extreme data sparsity conditions. Thus, in real-world cold-start scenarios, integrating content-based signals or user profile enrichment may be essential. This training stability further validates the model's readiness for practical recommendation deployment under sparse data conditions [14].

3.5. Product Recommendation Analysis

Table 5 presents concrete examples of product recommendations generated by RBM for two user samples, illustrating the practical effectiveness of this model. examples of product recommendations generated by RBM for two users based on their purchase history. generated for several users based on their initial product interactions.

Table 5. Sample RBM-Based Product Recommendations

User ID	Purchased Products (Sample)	RBM Recommendations (Top 3)
1	A-D-0.2, R-M-0.5	A-L-1, L-L-2.5, E-D-1

2	L-D-2.5, R-L-0.2	A-M-0.5, E-M-1, R-D-0.2

The RBM generated recommendations were contextually more relevant than those from the baseline models. This table provides concrete examples of the RBM's personalization capabilities. It shows how the model generates unique recommendations for two different users based on their distinct purchase histories. For example, the recommendations for User 1 are entirely different from those for User 2, reflecting the model's ability to capture individual preferences rather than relying on general popularity. This demonstrates the system's practical output and highlights its superiority over non-personalized baseline methods, which would have offered the same popular items to both users.

3.6. Heatmap Analysis of Latent Represetation

This heatmap illustrates how different products cluster around specific hidden units, revealing the model's capacity to group semantically related items. The y-axis represents the unique product IDs, and the x-axis represents the hidden units. The color intensity indicates the weight strength, revealing latent relationships learned by the RBM.

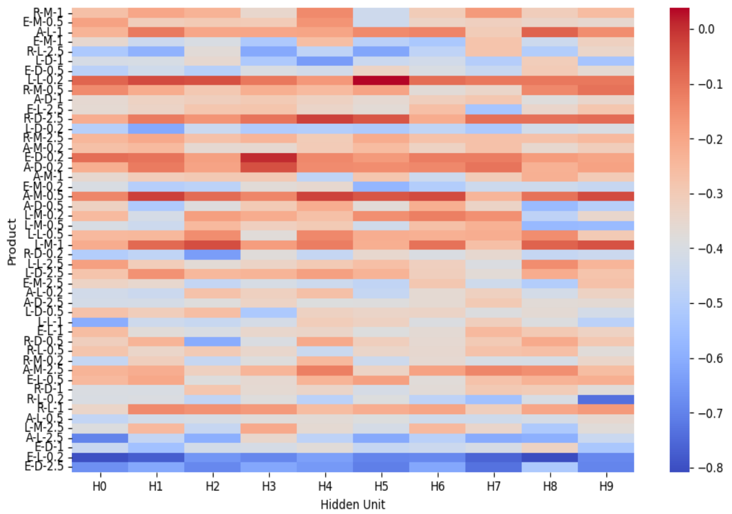

Fig. 4. Weight Heatmap of the Connections between Products and the 10 Hidden Units

The heatmap reveals that products with strong weight connections to the same hidden unit are likely to be recommended together. For example, as shown in Fig 4, products A-D-0.5 and A-M-1 both exhibit strong positive connection weights (dark red) with Hidden Unit 4 (H4). This indicates that the model has learned a latent feature, where users who purchase product A-D-0.5 are likely also to be interested in A-M-1. This pattern suggests that the two products may share the similar hidden attributes (e.g., flavor profile or specific roasting level) captured by H4. This insight cannot be found using simple popularity methods alone. This clustering effect suggests that the RBM has learned meaningful latent structures, which effectively represent underlying relationships among products. Such representations are crucial for improving personalization in recommendation systems.

In contrast to the strong positive clustering seen in H4, other hidden units learned different types of relationships. For example, Hidden Unit 7 (H7) exhibits a distinct pattern, characterized by strong negative weights (dark blue) for certain products, such as 'R-L-0.2' and 'A-L-2.5'. This suggests that H7 may have learned a differentiating latent feature, such as a specific flavor profile (e.g., highly acidic) or bean type that is distinct from others. A substantial negative weight indicates that products activating this feature are less likely to be co-purchased with products that do not. This ability to learn both associative (positive weights) and dissociative (negative weights) patterns further enriches the model's personalization capabilities. This varied learning across different hidden units reinforces the claim that the RBM is not merely memorizing but is building a meaningful, multi-faceted representation of user tastes.

3.7. Discussion

The experimental results provide several key findings regarding the application of RBM in a niche e-commerce cold-start scenario. First, the model architecture optimization reveals that a moderate number of hidden units (10 units) yields the best performance. This implies that for a dataset with high sparsity, a model with excessive capacity (e.g., 15 units) may lead to overfitting, whereas a model with too little capacity (e.g., 3 units) cannot adequately capture the complex latent patterns in user preferences. The reliability of these results is reinforced by our use of 5-fold cross-validation, which ensures the model was tested multiple times on different subsets of the data to provide a stable estimate of its performance, directly addressing the need for repeated testing to ensure suitability. One of the most critical findings of this study is the distinction between the model's reconstruction accuracy and its ranking accuracy. The significantly lower Mean Squared Error (MSE) compared to baseline methods demonstrates that RBM is highly effective at learning a user's overall taste profile, even with minimal data. However, the modest Precision@3 and NDCG@3 scores highlight that accurately predicting the exact top-3 items a user will purchase remains a significant challenge. This insight is fundamental for practical applications: the model can successfully identify a broad range of relevant products for a user, but its precision in top-N ranking is limited by data scarcity, a finding that becomes even more pronounced in the extreme cold-start user group.

Finally, the qualitative analysis via the weight heatmap (Fig. 4) confirms a crucial finding about the model's inner workings: the RBM is not merely memorizing purchases but is learning meaningful latent representations. The clustering of products around specific hidden units suggests that the model autonomously discovers implicit attributes (e.g., flavor profiles, bean origins) without any explicit labels. This capability is what allows the system to generate smarter and more personalized recommendations compared to simple popularity-based models.

3.8. Limitations and Future Research Directions

This study was conducted using a simulated dataset with a limited number of users and products, which may not fully reflect the complexities of a real-world e-commerce environment. While this study successfully established performance benchmarks using multiple metrics, the evaluation highlighted distinct levels of performance for different tasks. The model demonstrated strong reconstruction accuracy via MSE, but the initial results for ranking metrics (Precision@3, Recall@3, NDCG@3) were modest. These findings underscore the persistent challenge of top-N recommendation in cold-start scenarios and suggest that future work should focus on optimizing the model specifically for ranking-oriented tasks. The model also did not incorporate content-based attributes, such as product descriptions or coffee bean types, nor did it account for explicit user feedback [15]. Future research could provide a more robust evaluation by integrating real transaction data and expanding the scale of the dataset. The model could also be enhanced by combining RBM with content-based filtering techniques, thereby forming a hybrid approach that enhances recommendation diversity and accuracy [16]. Model performance can be further improved by comparing it with more modern deep learning architectures, such as Graph Neural Networks (GNN), which excel at modeling relationships between items, or Transformer-based models, which are now widely used for sequential recommendation tasks [17]. Testing the model on a live e-commerce platform would provide valuable insights into its real-world performance and adaptability. As this study demonstrates the feasibility of the Restricted Boltzmann Machine (RBM) algorithm, several limitations remain evident. The research employs a simulated dataset that may not fully represent the complex conditions of real-world e-commerce environments. The model omits essential content-based attributes, including product descriptions and direct user feedback, which may reduce its contextual accuracy. To strengthen future investigations, researchers should validate the model using large-scale transactional data from actual e-commerce operations. Such validation will enable a more accurate assessment of the model’s generalization capability. Researchers can further enhance model performance by integrating RBM with content-based filtering to develop a hybrid recommendation system. This hybrid approach could improve both the precision and diversity of product recommendations. To provide a more comprehensive performance benchmark, future studies should compare RBM with contemporary deep learning architectures, such as Graph Neural Networks (GNNs) and Transformer-based models, which are widely used for sequential recommendation tasks. These comparisons will help determine the RBM model’s relative effectiveness within modern recommendation frameworks.

4. Conclusion

This study demonstrates the successful application of a Restricted Boltzmann Machine (RBM) for generating personalized product recommendations in a simulated, niche e-commerce environment under extreme cold-start conditions. The methodological process yielded several key findings. First, the optimization process revealed a critical trade-off between model complexity and performance, with the 10-hidden-unit configuration providing the optimal balance between capturing latent user preferences and avoiding overfitting on the highly sparse dataset.

Furthermore, a significant finding from the evaluation process is the clear distinction between the model's strong reconstruction capability and its more modest top-N ranking accuracy. While the low MSE confirms that the RBM method is highly effective at understanding a user's general taste profile, the lower NDCG and Precision scores highlight the persistent challenge of high-precision ranking when historical data is minimal. This insight validates that while RBM can learn meaningful latent features, as confirmed by the heatmap analysis, its practical deployment in new marketplaces should ideally be part of a hybrid approach. Therefore, for future work, integrating content-based signals is highly recommended to enhance top-N recommendation quality. These findings validate RBM's capability in sparse-data environments and offer practical implications for early-stage e-commerce platforms seeking scalable and adaptive recommendation systems.

Acknowledgment

Thanks are addressed to official institutions or individuals who have provided funding or have made other contributions to the research. Acknowledgments are accompanied by a research contract number. This research receives full financial support from the DIPA Fund of Politeknik Negeri Sriwijaya for the Fiscal Year 2025 under Contract Number 08151/PL6.2.1/LT/2025, dated July 29, 2025. The study aims to deliver a measurable contribution to the advancement of science and technology, focusing specifically on the field of digital communication. The research outcomes are expected to strengthen academic understanding and provide practical insights for digital communication development.

Appreciation is also extended to the Telecommunications Engineering Laboratory of Sriwijaya State Polytechnic for providing facilities for the research. Special thanks also go to our MSME partners, namely the Cooperative and SME Agency of South Sumatra and Sriwijaya Coffee House, for their collaboration and valuable contributions throughout the research process.

Declarations

Author contribution.
Conceptualization: E.H., A.S.H., and S.
Methodology: M.Z.A. and E.R.
Software and experiments: M.Z.A., A.S.K., and L.A.
Validation and formal analysis: A.S.H., A., and M.A.
Writing – original draft preparation: E.H., S., and M.Z.A.
Writing – review and editing: A.S.H. and A.
Supervision and funding acquisition: A.S.H.
All authors have read and agreed to the published version of the manuscript.
Funding statement. This research was funded by the DIPA Fund of Politeknik Negeri Sriwijaya, grant number [08151/PL6.2.1/LT/2025] 2025.
Conflict of interest. The authors declare no conflict of interest.
Additional information. No additional information is available for this paper.

Data and Software Availability Statements

The dataset used and analyzed during the current study was simulated and is not publicly available due to its specific generation for this research context but is available from the corresponding author on reasonable request. The software code was developed using Python with the PyTorch library.

References

[1] J. David, E. Chapo, and S. Gustavo, "Intelligent systems with applications artificial intelligence and recommender systems in e-commerce . trends and research agenda," vol. 24, no. December 2023, 2024, doi: 10.1016/j.iswa.2024.200435.

[2] A.-M. Urdea and C. P. Constantin, “Exploring the impact of customer experience on customer loyalty in e-commerce,” in Proceedings of the International Conference on Business Excellence, vol. 15, no. 1, pp. 672–682, 2021, doi: 10.2478/picbe-2021-0063.
[3] N. Glisovic, D. Kragic, and M. Tegner, “Item cold start in e-commerce recommender systems: a survey,” IEEE Access, vol. PP, p. 1, 2025, doi: 10.1109/ACCESS.2025.3611026.
[4] H. Ko, S. Lee, Y. Park, and A. Choi, “A survey of recommendation systems: recommendation models, techniques, and application fields,” Electronics, vol. 11, no. 1, 2022, doi: 10.3390/electronics11010141.
[5] N. Hazrati and M. Elahi, “Addressing the new item problem in video recommender systems by incorporation of visual features with restricted boltzmann machines,” Expert Systems, vol. 38, no. 3, 2021, doi: 10.1111/exsy.12645.
[6] S. Wang, L. Cao, Y. Wang, Q. Z. Sheng, M. A. Orgun, and D. Lian, “A survey on session-based recommender systems,” ACM Computing Surveys, vol. 54, no. 7, 2023, doi: 10.1145/3465401.
[7] K. N. Mishra, A. Mishra, P. N. Barwal, and R. K. Lal, “Natural language processing and machine learning-based solution of cold start problem using collaborative filtering approach,” Electronics, vol. 13, no. 21, 2024, doi: 10.3390/electronics13214331.
[8] R. Alabdulrahman, “Towards personalized recommendation systems: Domain-driven machine learning techniques and frameworks,” 2020, [Online]. Available: http://dx.doi.org/10.20381/ruor-25236.
[9] Q. Xiaodong and L. A. N. Jiabao, “Research on improved rbm recommendation algorithm based on gibbs sampling,” Scalable Computing: Practice and Experience, vol. 26, pp. 1017–1034, 2025, doi: 10.12694/scpe.v26i3.4166.
[10] M. Benleulmi, I. Gasmi, N. Azizi, and N. Dey, “Explainable AI and deep learning models for recommender systems: State of the art and challenges,” Journal of Universal Computer Science, vol. 31, no. 4, pp. 383–421, 2025, doi: 10.3897/jucs.122380.
[11] I. Technology, “Beyond accuracy: Understanding and modeling the role of user conformity in recommender systems,” no. March, 2024.
[12] A. Sharma, N. Patel, and R. Gupta, “Enhancing AI-powered recommendation engines using collaborative filtering and neural network-based algorithms,” pp. 1–23, 2023.
[13] P. Li, S. Azman, M. Noah, and H. M. Sarim, “A survey on deep neural networks in collaborative filtering recommendation systems,” in 2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC), 2022, pp. 121-126.
[14] M. Wang and A. P. J. Yu, “Advancing responsible recommendation systems: Enhancing accuracy, diversity, and fairness,” no. May, 2025.
[15] R. J. Kuo and J. T. Chen, “An application of differential evolution algorithm-based restricted boltzmann machine to recommendation systems,” pp. 701–712, doi: 10.3966/160792642020052103008.
[16] D. K. Behera, M. Das, S. Swetanisha, and P. K. Sethy, “Hybrid model for movie recommendation system using content K-nearest neighbors and restricted boltzmann machine,” Indonesian Journal of Electrical Engineering and Computer Science, vol. 23, no. 1, pp. 445–452, 2021, doi: 10.11591/ijeecs.v23.i1.pp445-452.
[17] B. Kim et al., “CUPID: A real-time session-based reciprocal recommendation system for a one-on-one social discovery platform,” in IEEE International Conference on Data Mining Workshops (ICDMW), pp. 704–713, 2024, doi: 10.1109/ICDMW65004.2024.00097.
[18] G. S. Ananth, K. Raghuveer, and S. V. Kumar, “Preference weighted restricted boltzmann machines for cold start recommendations,” no. June, pp. 307–311, 2022.
[19] T. Arifin et al., “Pemanfaatan algoritma restricted boltzmann machines (RBM) untuk prediksi dini kanker paru-paru,” Jurnal Teknoinfo, vol. 5, no. 2, pp. 138–146, 2023, [in Indonesian].
[20] X. Cao, S. Chen, and Z. Li, “Research on e-commerce user behavior analysis based on big data collaborative recommendation algorithm,” vol. 5, pp. 30–37, 2021, doi: 10.23977/cpcs.2021.5100.
[21] E. S. Eriana and D. A. Zein, “Artificial intelligence,” Angewandte Chemie International Edition, vol. 6, no. 11, p. 1, 2023.
[22] D. C. Fatihah and I. S. Saidah, “Model promosi marketplace berbasis artificial intelligence

(AI) di Indonesia," JMBI UNSRAT (Jurnal Ilmiah Manajemen Bisnis dan Inovasi Universitas Sam Ratulangi), vol. 8, no. 3, pp. 806–817, 2021, doi: 10.35794/jmbi.v8i3.35908, [in Indonesian].

[23] F. H. J. Gesellschaft, I. Science, A. L. Reitbauer, J. Dat, B. Dipl, and M. Melcher, "Recommendation systems in (near) real-time," vol. D, 2023.

[24] N. Idrissi and A. Zellou, "A systematic literature review of sparsity issues in recommender systems," Social Network Analysis and Mining, vol. 10, no. 1, 2020, doi: 10.1007/s13278-020-0626-2.

[25] L. Liang, Z. Wen, Y. Shi, H. Liu, and W. Li, "Design of product recommendation system based on restricted boltzmann machine," vol. 6, no. 3, pp. 20–22, 2022.

[26] S. Malhotra, "Recommendation system based on restricted boltzmann machine," no. July, pp. 4000–4004, 2020.

[27] L. W. Member, X. H. Member, and X. W. Member, "A survey on accuracy-oriented neural recommendation: From collaborative filtering to information-rich recommendation," pp. 1–20, 2022.

[28] S. Lee, "The use of restricted boltzmann machines for clustering collaborative filtering," Journal of Autonomous Intelligence, vol. 6, no. 3, pp. 1–12, 2023, doi: 10.32629/jai.v6i3.723.

[29] A. Ibrahim et al., "A recommendation system for electric vehicles users based on restricted boltzmann machine and WaterWheel plant algorithms," IEEE Access, vol. 11, pp. 145111–145136, Nov. 2023, doi: 10.1109/ACCESS.2023.3345342.

[30] S. Sridhar, D. Dhanasekaran, and G. C. P. Latha, "Content-based movie recommendation system using MBO with DBN," Intelligent Automation & Soft Computing, vol. 35, no. 3, pp. 3241–3257, 2023, doi: 10.32604/iasc.2023.030361.