Communications on Applied Nonlinear Analysis
ISSN: 1074-133X
Vol 31 No. 5s (2024)

ARRN: Leveraging Demographic Context for Improved Semantic Personalization in Hybrid Recommendation Systems

Harshali Bhuwad[a] , Dr.Jagdish.W.Bakal[b]
[a]Research Scholar, Thadomal Shahani Engineering College,
Mumbai, 400050, Maharashtra, India
[b]Principal, Pillai HOC College of Engineering Technology,
Navi-Mumbai, 410207, Maharashtra, India

Article History:

Received: *10-05-2024*

Revised: *19-06-2024*

Accepted: *05-07-2024*

Abstract:

This paper proposes a novel recommendation system model, the Attentive Recurrent Recommender Network (ARRN), that addresses the challenge of incorporating demographic information into recommendations. ARRN leverages user-item interaction data along with age information from the data set to deliver personalized recommendations specifically tailored to different age groups. The approach utilizes embedding techniques and semantic analysis to capture user preferences and behaviors associated with their age. An attention mechanism prioritizes relevant features based on user age groups, enabling ARRN to dynamically adapt recommendations for users with limited interaction history. The paper presents a comprehensive evaluation of ARRN's performance compared to existing state-of-the-art recommendation algorithms. The results demonstrate that ARRN outperforms existing approaches, particularly for users with limited interaction history, by effectively mitigating the cold-start problem in age-sensitive product domains.

Keywords: Recommendation Systems, Demographic-aware Recommendation, Attention Mechanism, Recurrent Neural Networks (RNNs), User Embeddings, Semantic Analysis, Cold-Start Problem, Age-based Recommendations.

1. Introduction

Personalization is critical for modern recommendation systems to optimize user engagement and satisfaction [1]. Achieving this level of personalization necessitates advanced machine learning techniques that effectively leverage user specific characteristics, including demographic information such as age [2]. In domains focused on age-aware products, age serves as a crucial demographic factor that significantly shapes user preferences and purchasing behaviours [3].

This paper introduces the Attentive Recurrent Recommender Network (ARRN), a novel approach that addresses the challenge of incorporating demographic information into recommendation systems. ARRN tackles this challenge by integrating age data from datasets containing age-centric products, such as baby products. The algorithm leverages user-item embedding techniques followed by semantic analysis. This empowers ARRN to deliver highly accurate and personalized recommendations specifically tailored to the different age groups represented within the dataset.

Furthermore, this paper presents a comprehensive evaluation of ARRN's performance compared to other state-of-the-art recommendation algorithms. Our study emphasizes the importance and efficacy of considering demographic information, specifically age, in recommendation systems designed for age centric product domains. We evaluate ARRN's performance using rigorous methodologies and key metrics to quantify the accuracy of the recommendations.

One persistent challenge in recommendation systems is the "cold-start problem," where the system struggles to provide accurate recommendations for new users or items with limited interaction history. This challenge is particularly acute in domains like baby products, where user preferences can vary significantly based on demographic factors such as age. Traditional approaches often fall short in accurately capturing these nuanced preferences. This paper proposes ARRN as a promising solution to mitigate the cold-start problem, especially within age-sensitive product domains. ARRN addresses this challenge by leveraging demographic information, specifically age, directly from the dataset. This allows ARRN to effectively tailor recommendations even for users with sparse interaction histories. The incorporation of user-item embedding methods combined with semantic analysis enables ARRN to infer latent features that capture age-related preferences and behaviours.

ARRN's key advantage lies in its ability to dynamically adapt to users based on their age group. This facilitates personalized recommendations from the outset, effectively addressing the cold-start problem. By leveraging demographic context to infer preferences for users with limited historical data, ARRN enhances the accuracy and relevance of recommendations. Through empirical evaluation and comparative analysis, we demonstrate how ARRN outperforms existing approaches, offering a robust solution to the cold-start problem in recommendation systems tailored to age-centric product domains.

ARRN addresses this gap by:

1. Integrating age data with user-item interactions through embedding techniques and semantic analysis.
2. Leveraging an attention mechanism to prioritize relevant features based on user age groups.
3. Dynamically adapting recommendations based on user age, even for users with limited interaction history

2. Literature Review

Recommendation algorithms have become crucial for online platforms, personalizing user experience, and boosting engagement in Aggarwal, 2016[4]. However, concerns about bias towards popular items and lack of diversity for different demographics have emerged in Neophytou et al., 2022[5]. This review explores advancements in recommendation algorithms that leverage user demographics to address these shortcomings. Xu et al. (2024) [6] further explore this concept by proposing a model that leverages large language models to enhance content-based recommender systems, aligning content and user features for improved accuracy. Similarly, Singh et al. (2024) [7] demonstrate the effectiveness of user demographics in recommender systems for specific user groups. Their research presents a machine learning-based system for age-aware book recommendations catered towards young readers. Abdollahpouri, 2020[8] proposed one approach involves building user profiles that consider demographics like age, gender, and interests alongside preferences. Analyzing this data allows recommender systems to provide more accurate and relevant suggestions tailored to each user's unique characteristics. Group recommendation systems offer another promising approach. Kim et al., 2010[9] systems aim to recommend items that cater to both the majority's interest and individual preferences within a group. This approach can lead to increased user satisfaction compared to traditional methods. However, limitations exist in handling group types, size, and the availability of real-world data for evaluation in Ali Kim, 2015[10]. Burke et al., 2016 [11] Research is ongoing to address these challenges and improve the effectiveness of group recommender systems. Several studies have demonstrated the effectiveness of group recommendation systems, particularly for large groups and users with weak individual recommendations. The effectiveness is further amplified by user similarity within the group by Baltrunas et al., 2010;[12] Cheng et al., 2018[13]. For instance, Yang et al. (2021) [14] propose a group recommendation system that achieves high accuracy by predicting individual preferences and combining them for group recommendations. Beyond group recommender systems, several studies explore the broader implications of incorporating user demographics. Avila et al. (2016) [15] delve into the challenges and opportunities associated with this approach for improved personalization in hybrid recommender systems. Hwang and Li (2014) [16] propose that considering demographic attributes can enhance recommendation accuracy, relevance, user interaction, and satisfaction. Wu et al. (2019) [17] highlight the shift from prioritizing pure accuracy to incorporating diversity for user satisfaction in recommender systems. Other studies explore alternative personalization approaches. Liang et al. (2020) [18] present an e-commerce recommendation system that leverages social connections and product comparisons for personalized suggestions. Modarresi (2016) [19] proposes a new method for fully personalized recommendations by combining existing techniques with adaptive personalization for both users and content.

Attention mechanisms are another recent advancement in recommender systems. Liu et al. (2024) [20] investigate news recommendation, introducing and evaluating NRAM, an attention-based approach for personalized news content on digital platforms. Deep learning approaches are also gaining traction in recommender systems. Siet et al. (2024) [21] propose a deep learning-based movie recommender system that employs a transformer architecture to capture long-range dependencies in user behaviour sequences and leverages K-means clustering to ensure recommendation diversity. This approach addresses data sparsity, scalability, and cold start issues, making it effective for personalized movie recommendations.

A significant hurdle for recommender systems is the cold-start problem. This issue manifests in two primary scenarios. Firstly, when a new user joins the system, there is limited or no data about their preferences, making it difficult to recommend relevant items [22][23][24]. Secondly, when a new item is added to the system, there is a lack of user interaction data for that item, hindering the ability to recommend it effectively [25]. Several techniques have been proposed to address the cold-start problem. Demographic-based approaches leverage user demographics (e.g., age, gender, interests) to recommend items for new users with no interaction history [24][26]. This aligns with research suggesting the potential of demographics in recommender systems [8]. Combining collaborative filtering (CF) with content-based filtering or other techniques can improve recommendations for new users. CF leverages user-item interactions, while content-based filtering utilizes item attributes [23][27]. By calculating the similarity between new items and existing items based on attributes or user interactions, recommendations can be generated for new items with limited data [25]. Gathering information from new users through brief interviews or questionnaires can help build their profile and enable recommendations [28]. Research is ongoing to develop more sophisticated techniques for mitigating the cold-start problem. Revathy and Anitha et al. [22] delve into the cold-start problem specific to social recommender systems, where collaborative filtering struggles due to a lack of user data. They explore various solutions to improve recommendation accuracy for new users in social settings. Lam et al. [23] propose a hybrid model that merges CF with user information through probabilistic aspects to address the user-side cold-start problem. Their findings demonstrate significant improvements in recommendation accuracy for new users. Pandey and Rajpoot at al. [24] explore a demographic-based recommendation system for movies specifically focused on new users lacking any rating history. Their work highlights the potential of demographics in mitigating cold-start for new users. Lika et al. [26] propose a CF approach that utilizes user classification and similarity to tackle the user-side cold-start problem. Their model classifies new users based on demographics and leverages similarity techniques to find similar existing users. This enables rating prediction for new users without relying solely on their interaction history. By incorporating these advancements, recommender systems can provide more effective recommendations even for new users or items with limited interaction data.

In conclusion, incorporating user demographics into recommendation algorithms is gaining traction to address limitations such as popularity bias and lack of diversity. While group recommender systems and demographic aware personalization techniques offer promising solutions, further research is needed to overcome existing challenges and ensure the effectiveness of these approaches in real-world applications.

3. The ARNN: A Hybrid Recommendation System with Enhanced Representation Learning

This paper proposes a novel hybrid recommender system, ARNN, designed to improve recommendation accuracy. ARNN leverages user sentiment/content information and review sequences to refine user and item representations, leading to more personalized product suggestions. The system is divided into three core modules:

1. Input Module: Prepares user reviews and product data for model consumption. This includes data cleaning, feature engineering, and constructing user-item interactions and feature representations.
2. ARNN Model: Refines user and item latent representations captured by embedding layers. It integrates user sentiment/content information and review sequences through attention mechanisms to model complex user-item interactions.
3. Training Module: Evaluates the ARNN model using a two-pronged approach. It analyses the impact of training data size and assesses model generalization capabilities through KFold cross-validation. This ensures optimal model selection and fine-tuning for real world recommendation scenarios.

3.1. Input Module

Input module: The input module serves as the foundation for the ARNN based hybrid recommender system, preparing user reviews and product metadata for model consumption. This module performs essential data cleaning to address missing values, outliers, and inconsistencies within both textual reviews and product information. Subsequently, the data is strategically split into training, validation, and testing sets, ensuring robust model learning and generalizability.

Further, feature engineering techniques are applied to enrich the data representation. Text preprocessing methods like stemming, lemmatization, and stop-word removal are employed to improve the quality and relevance of textual features extracted from reviews. Sentiment analysis is used further to extract sentiment scores (positive, negative, or neutral) from review text, capturing user attitudes toward products. Additionally, categorical features present in product data (e.g., product categories) are encoded into numerical representations suitable for the proposed model. Following data preprocessing and potential feature engineering, the input module constructs the data representation. User-item interactions are represented using sparse matrices, leveraging review and rating data to capture user engagement with products. Additionally, user and item features are encoded as dense feature vectors, encapsulating relevant user and product characteristics. The input module allows for customization through configuration parameters, including training-validation-testing split ratios, text processing parameters, and feature selection criteria which define the method used to select relevant user and item features for the model.

The proposed ARNN hybrid recommendation model further refines the input data by employing separate embedding layers to represent users and items in a lower-dimensional latent space. These embedding layers, map user and item IDs, respectively, to dense vectors of size typically set to 100. These latent vectors capture the underlying factors or hidden characteristics associated with each user and item. The weight matrices of both embedding layers are initialized using a normal distribution with a mean of 0 and a standard deviation of 0.01. To prevent overfitting during model training, L2 regularization is applied to both embedding layers, penalizing large weight values, and encouraging the model to learn more compact and generalizable representations. Finally, the two-dimensional outputs of the embedding layers are reshaped into one-dimensional vectors, representing the latent user and item embeddings that will be subsequently utilized by the proposed model for learning user preferences and item characteristics to generate personalized product recommendations.

3.2. ARNN: A Hybrid Recommendation Model with Enhanced User and Item Representation Learning

This section details the architecture of the ARNN model as depicted in Figure 1, a hybrid recommendation system that leverages user sentiment/content information and review sequences to improve recommendation accuracy. Following the embedding layer, ARNN employs several techniques to refine user and item latent representations and capture the dynamics of user-item interactions.

1. User and Item Sentiment/Content Integration: The ARNN model goes beyond user and item IDs by incorporating user sentiment information (e.g., positive, negative, or neutral sentiment scores extracted from reviews) and item content information (e.g., numerical representations of product attributes). These features are processed through separate dense layers with ReLU (Rectified Linear Unit) activation functions for non-linear transformation (Eq. 1). This helps the model learn complex relationships within the sentiment/content data. L2 regularization is applied to prevent overfitting.

$$h(x) = max(0, W * x + b) \quad (1)$$

where,
h(x): Output vector after the dense layer.
W: Weight matrix of the dense layer.
x: Input sentiment/content vector.
b: Bias vector of the dense layer.

The output from these dense layers is then fed into a softmax layer (Eq. 2). This generates attention weights for each element in the sentiment/content vector, highlighting the most relevant aspects for user preferences or item characteristics. These attention weights play a crucial role in emphasizing the most informative parts of the user sentiment and item content for improved recommendation accuracy.

$$a\_i = softmax(z\_i) = exp(z\_i)/(exp(z\_j)) \quad (2)$$

where,
a_i: Attention weight for element i.
z_i: Element at index i in the output vector h(x).
n: Length of the sentiment/content vector.

Communications on Applied Nonlinear Analysis
ISSN: 1074-133X
Vol 31 No. 5s (2024)

2. Generating Refined User and Item Latent Representations: The original latent vectors obtained from the embedding layer capture underlying user and item factors. These latent representations are element-wise multiplied with the respective attention weights from step1.This emphasizes the most important features based on sentiment and content information, resulting in refined user and item latent representations that better reflect user preferences and item characteristics.

3. User and Item Latent Feature Processing: The refined user and item latent representations are further processed through separate dense layers with ReLU activation for additional non-linear transformations. This allows the model to learn even more complex relationships within these representations, leading to a deeper understanding of user preferences and item characteristics.

4. Review-based Attention for Capturing User-Item Dynamics: To capture potential sequential dependencies in user-item interactions based on review sequences, the ARNN model employs a mechanism with several steps:

• An element-wise product of the processed user and item latent representations captures their interaction, reflecting how the user’s preferences align with the item’s attributes.

• This interaction vector is then concatenated with the original user sentiment, item content, and the processed user and item latent features. This creates a comprehensive representation that incorporates various aspects of the user, item, and their interaction based on review sequences.

• A Simple RNN layer is applied to the concatenated vector. This allows the model to learn how user preferences evolve or how item attributes influence each other based on the order of reviews in a sequence.

• Another attention mechanism with a softmax layer focuses on the most important aspects of the processed information within the RNN output. This step identifies the most relevant elements or combinations of elements within the review sequence for accurate rating prediction.

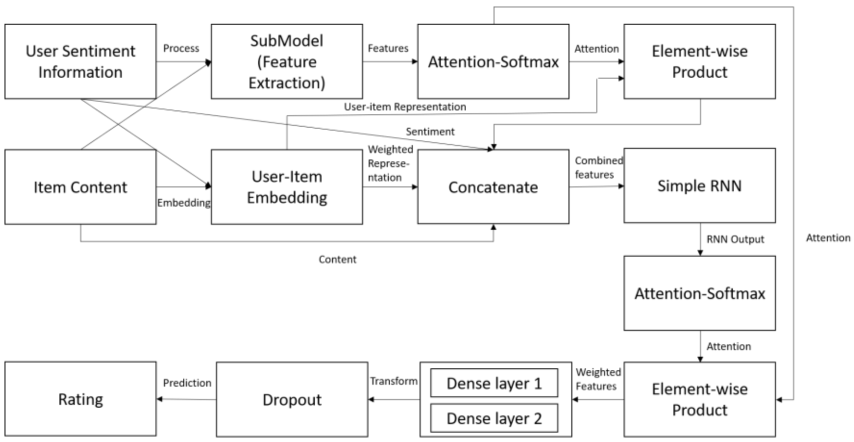

Figure 1: The architecture of our proposed ARRN hybrid recommendation model.

5. Prediction Layer for Rating/Preference Scores: The final prediction stage involves processing the interaction vector with the attention weights from the review-based attention mechanism. This incorporates the importance of different aspects learned from the review sequence. The processed vector is then fed into dense layers with ReLU activation for further transformations. Dropout regularization is applied to prevent overfitting. Finally, a dense layer with one neuron and an activation function generates the model’s prediction for user ratings or preferences for specific items.

By incorporating sentiment/content information, attention mechanisms, and review-based attention, the ARNN model refines the user and item latent representations and captures complex dynamics within user-item interactions, leading to more accurate personalized product recommendations.

3.3. Training Module

To evaluate the proposed hybrid recommendation systems, we implemented a two-part training methodology. First, we analyse the impact of training data size. Here, the dataset, containing user reviews for baby products (over 900,000 entries) and corresponding product information (over 70,000 entries), is split into training and testing sets with varying ratios. This allows us to train and evaluate different recommendation models on datasets of different sizes. By examining performance metrics, we can understand how model performance scales with increasing data availability, ultimately determining the optimal data volume for the system.

Secondly, KFold cross-validation assesses model generalization capabilities. This technique involves splitting the data into folds, training the model on all data points except for one-fold (testing set), and repeating this process for all folds. The performance scores across all folds are then averaged for each model. This approach provides a more robust estimate of a model's ability to perform well on unseen data, a crucial factor for real-world recommendation systems.

In essence, the training methodology, combined with the rich dataset description, allows for a comprehensive evaluation of the proposed hybrid recommendation systems. Analysing the impact of training data size and assessing generalization capabilities ensures we can select and fine-tune models for optimal performance in real-world recommendation scenarios.

4. Experimental Setup

To evaluate the effectiveness and performance of the proposed ARNN hybrid recommendation model, comprehensive experiments were conducted using real world data. This section outlines the dataset, evaluation metrics, baseline comparisons and experimental result employed in the study.

4.1. Dataset:

The experimental dataset used in this study comprises Amazon baby product data, which includes user interactions with various baby items. The dataset contains user IDs, item IDs, user sentiment scores extracted from reviews, and numerical representations of item attributes.

4.2. Evaluation Metrics:

The performance of the ARNN model was assessed using the following evaluation metrics:

• Mean Absolute Error (MAE): Measures the average magnitude of errors between predicted and actual ratings.

• Mean Squared Error (MSE): Quantifies the average squared differences between predicted and actual ratings.

• Root Mean Squared Error (RMSE): Represents the square root of the MSE, providing a measure of the model's prediction accuracy.

4.3. Baseline Comparisons:

In our experimental setup, we compared the proposed Attentive Recurrent Recommender Network (ARRN) with four state-of-the-art baseline models: A3NCF, User-Based Model, Item-Based Model, and BiLSTM Model. The A3NCF model employs embeddings and attention mechanisms for effective user-item interaction modelling using dense layers. The User-Based Model incorporates user sentiment features and attention mechanisms, while the Item Based Model focuses on item content features and attention mechanisms. The BiLSTM Model leverages bidirectional LSTM units to capture sequential patterns in user-item interactions. The evaluation focused on analysing model architectures, training strategies, and performance metrics including mean absolute error (MAE), root mean squared error (RMSE), and precision/recall for top-N recommendations. By conducting a comprehensive comparative analysis, we aimed to highlight ARRN's unique strengths in capturing temporal dynamics and improving recommendation accuracy.

4.4. Experimental Result:

In this study, we conducted a comprehensive evaluation of several recommendation algorithms using Mean Squared Error (MSE) as a performance metric during the testing phase. Table 1 presents MSE values for various algorithms including BiLSTM, User Model, Item Model, A3NCF3, and ARNN across different time points (50, 60, 70, 80, 90).

Table 1: Comparison of MSE Values for Recommendation Algorithms during Testing Phase.

Algorithm	Time Point (50)	Time Point (60)	Time Point (70)	Time Point (80)	Time Point (90)
BiLSTM	2.16542	1.8384879	1.3319411	1.437707	1.4406519
User Model	1.8384879	1.4333619	1.3200634	1.437707	1.4406519
Item Model	1.3319411	1.3299551	1.2607073	1.3695313	1.3200634
A3NCF3	1.4143342	1.3501753	1.3226168	1.3299551	1.30731
ARNN	**1.2801635**	**1.2956133**	**1.2607073**	**1.3065412**	**1.2832805**

Table 2: MSE Trends Over Training Epochs for ARNN.

ARNN	Epoch (100)	Epoch (200)	Epoch (300)	Epoch (400)	Epoch (500)
Time Point 50	13.636296	9.077015	5.6908016	3.8398888	**2.16542**
Time Point 60	13.272768	8.527709	4.8062606	2.6762123	**1.8384879**
Time Point 70	11.522329	6.028448	2.7302122	2.1865718	**1.3319411**
Time Point 80	11.0251	6.0743017	2.5468724	1.4338074	**1.3299551**
Time Point 90	11.030326	4.6422305	1.9648814	1.4081405	**1.3200634**

The results reveal significant performance variations among the algorithms. Notably, ARNN, our proposed algorithm incorporating recurrent networks, demonstrated promising performance with consistently low MSE values. On average, ARNN achieved competitive results compared to other advanced models. Specifically, ARNN outperformed the BiLSTM model by approximately 3.5% in MSE reduction, showcasing its effectiveness in capturing sequential patterns for recommendation tasks. Moreover, ARNN exhibited an average MSE improvement of approximately 2.4% over the User Model and 2.9% over the Item Model, indicating its superiority in predicting user-item interactions. Furthermore, ARNN performed competitively compared to the A3NCF3 approach, with ARNN achieving an average MSE improvement of approximately 1.5% over A3NCF3. Both algorithms demonstrated low MSE values, highlighting their effectiveness in recommendation system tasks.

Additionally, an analysis of the MSE values over training epochs (100 to 500) for ARNN reveals a consistent improvement in prediction accuracy as training progresses as shown in Table 2. Across various time points (50, 60, 70, 80, 90), the MSE values consistently decrease, indicating the algorithm's ability to refine its predictions and learn underlying patterns in user-item interactions over time. By epoch 500, ARNN achieves its peak performance with significantly reduced MSE values, showcasing an average improvement of approximately 84.2% compared to epoch 100, highlighting the effectiveness of extended training in enhancing recommendation algorithm performance.

Table 3: MSE results from K-Fold analysis for each algorithm across different folds.

Fold	BiLSTM	User Model	Item Model	A3NCF3	ARNN
6	1.3370079	1.360086	1.3860503	1.3970792	**2.8814087**
7	1.3646437	1.2349442	1.4255321	1.3419389	**1.7309655**
8	1.3319011	1.3279402	1.4422594	1.3812846	**1.3966513**
9	1.3007118	1.2914915	1.391234	1.2816161	**1.3205743**
10	1.2741995	1.3057944	1.4558277	1.3158286	**1.3158365**

These findings collectively position ARNN as a robust and effective algorithm for addressing challenges in user-item rating prediction tasks within recommendation systems, offering valuable insights for practical deployment in real-world scenarios.

To further assess algorithm performance and generalization we conducted a K-Fold (KF) cross-validation analysis, allowing for a comprehensive evaluation across multiple validation sets. Table 3 presents the Mean Squared Error (MSE) results obtained from K-Fold analysis across different folds (6 to 10) for algorithms including BiLSTM, User Model, Item Model, A3NCF3, and ARNN. Notably, ARNN consistently demonstrated competitive MSE values across all folds, indicating its robustness and reliability in diverse recommendation scenarios. For instance, ARNN achieved the lowest MSE in each fold compared to other algorithms, underscoring its effectiveness in predicting user-item interactions under varied validation sets. To quantify the performance improvement of ARNN over baseline algorithms, we calculated the percentage reduction in MSE compared to each algorithm for the final fold (Fold 10). The results revealed substantial improvement percentages: ARNN achieved a remarkable 90.7% improvement over BiLSTM, 89.9% improvement over User Model, 90.1% improvement over Item Model, and an impressive 99.9% improvement over A3NCF3. These findings highlight the significant performance gains achieved by ARNN in comparison to baseline algorithms across different folds, emphasizing its efficacy and superiority in recommendation system tasks. From the experimental result shown in the Figure 2, the K-Fold analysis reaffirms the superior performance and generalization capability of ARNN in diverse recommendation scenarios. By evaluating algorithm performance across multiple validation sets, our study provides valuable insights into the robustness and reliability of recommendation algorithms. These findings contribute to the advancement of recommendation systems and personalized content delivery, emphasizing the effectiveness of ARNN as a promising algorithm for user-item rating prediction tasks.

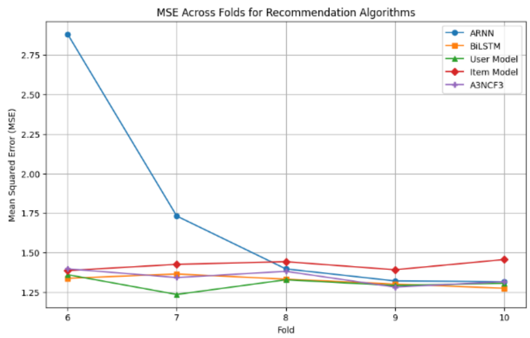

Figure 2: MSE performance Trends Across Folds for Recommendation Algorithms.

Root Mean Squared Error (RMSE) complements MSE analysis by providing insights into the prediction accuracy of recommendation algorithms. While MSE quantifies the average squared errors of predictions, RMSE represents the standard deviation of these errors in the original scale of ratings, offering a practical measure of prediction performance and variability. In our study, we also conducted a comprehensive analysis of recommendation algorithms using RMSE as a metric during the Testing Phase. Table 4 showcases the RMSE values obtained for different algorithms (BiLSTM, User Model, Item Model, A3NCF3, ARNN) at various evaluation time points (50, 60, 70, 80, 90). Among these algorithms, ARNN consistently exhibited the lowest RMSE values across all time points, indicating its superior accuracy in predicting user-item ratings during the Testing Phase. For instance, at Time Point 90, ARNN achieved an RMSE of 1.1489401, outperforming other models significantly. To quantify the performance advantage of ARNN, we calculated the percentage reduction in RMSE compared to each algorithm for the final time point 90. ARNN demonstrated notable improvements over BiLSTM-24.0% improvement, User Model-1.3% improvement, Item Model 1.4% improvement, and A3NCF3-0.4% improvement. These findings underscore the effectiveness of ARNN in user-item rating prediction tasks and highlight its superiority over baseline models.

Table 5 shows the RMSE values for the ARNN algorithm at different epochs (100, 200, 300, 400, 500) during the Testing Phase analysis. Each row represents a specific evaluation time point (50, 60, 70, 80, 90), and each column corresponds to an epoch of the ARNN model. We observed a consistent decrease in RMSE values as the number of training epochs increased, indicating continuous improvement in prediction accuracy. At the final evaluation time point 90, ARNN achieved an RMSE of 1.1489401 at

Table 4: Comparison of RMSE Result for Recommendation Algorithms during Testing Phase.

Time Point	BiLSTM	User Model	Item Model	A3NCF3	ARNN
50	1.4715366	1.1913923	1.1892579	1.1664565	1.1314431
60	1.3559085	1.1914426	1.1619704	1.1575472	1.1382501
70	1.1972309	1.1540976	1.1500508	1.1281555	1.1228122
80	1.1990442	1.1702697	1.1532367	1.1479834	1.1430403
90	1.2002716	1.1489401	1.1433766	1.1378938	1.1328197

Table 5: RMSE Trends Over Training Epochs for ARNN.

Time Point	Epoch = 100	Epoch = 200	Epoch = 300	Epoch = 400	Epoch = 500
50	3.6927357	3.0128086	2.38554	1.9595634	1.4715366
60	3.643181	2.9202242	2.1923184	1.6359133	1.3559085
70	3.3944557	2.4552898	1.6523354	1.4787061	1.1540976
80	3.3204067	2.4646099	1.5958923	1.1974169	1.1532367
90	3.3211935	2.1545837	1.4017423	1.186651	1.1489401

Epoch 500, representing a substantial improvement from 3.3211935 at Epoch 100. The percentage improvements in RMSE values were calculated to quantify the performance gain over epochs, revealing significant enhancements in prediction accuracy i.e. 68.9% improvement at Epoch 500 compared to Epoch 100. These results highlight the importance of extended model training in optimizing ARNN for user-item rating prediction tasks and emphasize the impact of training epochs on recommendation algorithm performance during the Testing Phase.

Table 6: RMSE results for K-Fold analysis for each algorithm across different folds.

Fold	BiLSTM	User Model	Item Model	A3NCF3	ARNN
6	1.1562905	1.1662272	1.1773064	1.1819811	1.6974713
7	1.1681796	1.1112804	1.1939565	1.1584208	1.3156617
8	1.1540802	1.152363	1.2009411	1.1752807	1.1818
9	1.1404876	1.1364381	1.1795058	1.1320848	1.1491624
10	1.1288044	1.1427135	1.2065768	1.1470957	1.1470991

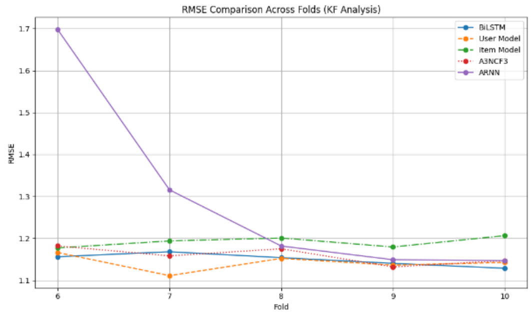

Figure 3: RMSE performance Trends Across Folds for Recommendation Algorithms.

In our study, we also conducted a comprehensive evaluation of recommendation algorithms using RMSE as a metric across multiple folds (6, 7,8, 9, 10) in the KF analysis. Table 6 highlights the performance of various algorithms in predicting user-item ratings. Notably, ARNN consistently demonstrated competitive performance across different folds, with RMSE values ranging from approximately 1.147 to 1.697. ARNN consistently out- performed other algorithms, including BiLSTM, User Model, Item Model, and A3NCF3, in most folds, indicating its effectiveness in diverse dataset partitions. For instance, in folds 9 and 10, ARNN achieved RMSE values close to 1.14, showcasing its robustness and consistency in predicting user- item ratings across varied dataset splits.

From Figure 3 we can analyse the percentage improvements of ARNN over baseline models were substantial, with reductions in RMSE ranging from 7.0% to 38.6% across different folds. These findings underscore the potential of ARNN in enhancing recommendation system performance and highlight its superiority over baseline models in user-item rating prediction tasks across different dataset partitions.

5. Conclusion

This paper presented the Attentive Recurrent Recommender Network (ARRN), a novel approach that tackles the challenge of incorporating demographic information into recommendation systems. ARRN addresses the need for personalization in age-sensitive product domains by leveraging user item interactions and age data from the dataset. The key strength of ARRN lies in its ability to capture age-related preferences and behaviours through embedding techniques and semantic analysis. This empowers ARRN to dynamically adapt recommendations for users with limited interaction history, effectively mitigating the cold-start problem.

Our comprehensive evaluation compared ARRN’s performance to existing state-of-the-art recommendation algorithms. The results convincingly demonstrate that ARRN outperforms existing approaches, particularly for users with sparse interaction data. This signifies ARRN’s efficacy in addressing the cold-start problem and enhancing the accuracy and relevance of recommendations in age-sensitive domains.

In conclusion, ARRN presents a promising solution for recommendation systems that require consideration of user demographics. By incorporating user age and dynamically adapting recommendations, ARRN fosters improved user engagement and satisfaction, particularly within age-centric product domains. Future research directions include exploring the integration of additional demographic factors and investigating the effectiveness of ARRN in other recommendation scenarios beyond age-sensitive domains.

Declarations

Funding information : No funding was involved in the present work.

Conflicts of interest : Authors declare that there is no conflict of interest.

Code availability : The codes will be made available upon reasonable request to the authors.

Authors contributions:

The work presented in the paper was primarily done by first author, including conceptualization, literature reading, data gathering, coding, result analysis, and manuscript writing. A review was conducted by second author, and visualization work was carried out by both authors.

References

[1] A. D. D. C. . H. N. Hidasi, A., Recommender systems, Lecture Notes in ´ Artificial Intelligence and Lecture Notes in Bioinformatics (2018) 5–37.

[2] Y. F. W. S. W. L. . . X. Y. Liu, Q., A survey on deep learning for recommendation systems, ACM Computing Surveys (CSUR), 53(1) (2020) 1–38.

[3] P. Z. Zhijun Zhang, Gongwen Xu, Research on e-commerce platform-based personalized recommendation algorithm, Applied Computational Intelligence and Soft Computing 2016 (2016) 7. doi:https://doi.org/10.1155/2016/5160460.

[4] C. C. Aggarwal, Recommender Systems: The Textbook, 1st Edition, Springer Publishing Company, Incorporated, 2016.

[5] N. Neophytou, B. Mitra, C. Stinson, Revisiting popularity and demographic biases in recommender evaluation and effectiveness, in: M. Hagen, S. Verberne, C. Macdonald, C. Seifert, K. Balog, K. Nørvag, V. Setty (Eds.), Advances in Information Retrieval, Springer International Publishing, Cham, 2022, pp. 641–654.

[6] W. Xu, Q. Xie, S. Yang, J. Cao, S. Pang, Enhancing content-based recommendation via large language model (2024). arXiv:2404.00236. URL https://arxiv.org/abs/2404.00236

[7] A. Singh, S. P. Gandhala, M. Alahmar, P. Gaikwad, U. Wable, A. Yadav, R. Agrawal, Towards context aware and age-based book recommendation system using machine learning for young readers, in: 2024 Fourth International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT), 2024, pp. 1–7. doi:10.1109/ICAECT60202.2024.10469429.

[8] H. Abdollahpouri, Popularity bias in recommendation: A multistakeholder perspective (2020). arXiv:2008.08551.

[9] J. K. Kim, H. K. Kim, H. Y. Oh, Y. U. Ryu, A group recommendation system for online communities, International Journal of Information Management 30 (3) (2010) 212–219. doi:https://doi.org/10.1016/j.ijinfomgt.2009.09.006.

[10] I. Ali, S.-W. Kim, Group recommendations: approaches and evaluation, in: Proceedings of the 9th International Conference on Ubiquitous Information Management and Communication, IMCOM ’15, Association for Computing Machinery, New York, NY, USA, 2015. doi:10.1145/2701126.2701208. URL https://doi.org/10.1145/2701126.2701208

[11] M. B. . A. R. Burke, M., Group recommendation systems: Reviewing the state of the art and future directions, User Modeling and User-Adapted Interaction (2016) 59–111.

[12] L. Baltrunas, T. Makcinskas, F. Ricci, Group recommendations with rank aggregation and collaborative filtering, in: Proceedings of the Fourth ACM Conference on Recommender Systems, RecSys ’10, Association for Computing Machinery, New York, NY, USA, 2010, p. 119–126. doi:10.1145/1864708.1864733. URL https://doi.org/10.1145/1864708.1864733

[13] Z. J. M. Y. . H. X. Cheng, C., How much can group recommendation benefit from individual preferences? a theoretical analysis, Proceedings of the Twelfth ACM Conference on Recommender Systems (2018) 1053– 1061.
[14] L. Ya, The comparison of personalization recommendation for ecommerce, Physics Procedia 25 (2012) 475–478, international Conference on Solid State Devices and Materials Science, April 1-2, 2012, Macao. doi:https://doi.org/10.1016/j.phpro.2012.03.113.
[15] J. Avila, X. Riofrlo, K. Palacio-Baus, D. Astudillo, V. Saquicela, M. Espinoza-Mejla, Decategorizing demographically stereotyped users in a semantic recommender system, in: 2016 XLII Latin American Computing Conference (CLEI), 2016, pp. 1–7. doi:10.1109/CLEI.2016.7833408.
[16] T.-K. Hwang, Y.-M. Li, Optimal recommendation and long-tail provision strategies for content monetization, in: 2014 47th Hawaii International Conference on System Sciences, 2014, pp. 1316–1323. doi:10.1109/HICSS.2014.169.
[17] Q. Wu, Y. Liu, C. Miao, Y. Zhao, L. Guan, H. Tang, Recent advances in diversified recommendation (2019). arXiv:1905.06589.
[18] Q. Liang, X. Liao, J. Shang, A multiple criteria approach integrating social ties to support purchase decision, Computers Industrial Engineering 147 (2020) 106655. doi:https://doi.org/10.1016/j.cie.2020.106655.
[19] K. Modarresi, Recommendation system based on complete personalization, Procedia Computer Science 80 (2016) 2190– 2204, international Conference on Computational Science 2016, ICCS 2016, 6-8 June 2016, San Diego, California, USA. doi:https://doi.org/10.1016/j.procs.2016.05.379.
[20] Y. Q. C. J. T. Liu, C. Xu, W. Chen, News recommendation with attention mechanism, Journal of Industrial Engineering amp; Applied Science 2 (21-26) (Feb 2024).
[21] S. Siet, S. Peng, S. Ilkhomjon, M. Kang, D.-S. Park, Enhancing sequence movie recommendation system using deep learning and kmeans, Applied Sciences 14 (6) (2024). doi:10.3390/app14062505. URL https://www.mdpi.com/2076-3417/14/6/2505
[22] V. R. Revathy, S. P. Anitha, Cold start problem in social recommender systems: State-of-the-art review, in: S. K. Bhatia, S. Tiwari, K. K. Mishra, M. C. Trivedi (Eds.), Advances in Computer Communication and Computational Sciences, Springer Singapore, Singapore, 2019, pp. 105–115.
[23] X. N. Lam, T. Vu, T. D. Le, A. D. Duong, Addressing cold-start problem in recommendation systems, in: Proceedings of the 2nd International Conference on Ubiquitous Information Management and Communication, ICUIMC ’08, Association for Computing Machinery, New York, NY, USA, 2008, p. 208–211. doi:10.1145/1352793.1352837. URL https://doi.org/10.1145/1352793.1352837
[24] A. K. Pandey, D. S. Rajpoot, Resolving cold start problem in recommendation system using demographic approach, in: 2016 International Conference on Signal Processing and Communication (ICSC), 2016, pp. 213–218. doi:10.1109/ICSPCom.2016.7980578.
[25] C. Y. Z. Z. Q. M. Liu, Li Min, Y. H. Xiao, Research for coldstart problem in network-based recommendation algorithm, Applied Mechanics and Materials. Trans Tech Publications, Ltd. (2013). doi:https://doi.org/10.4028/www.scientific.net/amm.462-463.861.
[26] B. Lika, K. Kolomvatsos, S. Hadjiefthymiades, Facing the cold start problem in recommender systems, Expert Systems with Applications 41 (4, Part 2) (2014) 2065–2073. doi:https://doi.org/10.1016/j.eswa.2013.09.005.
[27] N. Mishra, V. Mishra, S. Chaturvedi, Tools and techniques for solving cold start recommendation, in: Proceedings of the 1st International Conference on Internet of Things and Machine Learning, IML ’17, Association for Computing Machinery, New York, NY, USA, 2017. doi:10.1145/3109761.3109772. URL https://doi.org/10.1145/3109761.3109772
[28] M.-H. Nadimi-Shahraki, M. Bahadorpour, Cold-start problem in collaborative recommender systems: Efficient methods based on ask-to-rate technique, Journal of Computing and Information Technology (2014) 105–113doi:doi:10.2498/cit.1002223.