HistoryofMedicine,2024,10(2):2241-2260
DOI:10.48047/HM.V10.I2.2024.2241-2260

A SYNERGISTIC APPROACH TO PRECISION RATING PREDICTION IN RECOMMENDER SYSTEMS USING DEEP LEARNING AND COLLABORATIVE FILTERING

Umer Farooq[1], Shahid Ameer[2*], Samreen Razzaq[3], Suleman Shahzad[3], Syed Sami Ahmad Bukhari[4], Anam Safdar Awan[3], Zainab Fatima[3], Amna Noor[4]

[1]Department of Software Engineering, University of Sargodha, Pakistan
[2]Department of Computer Science and Information Technology, Superior University Lahore, Pakistan
[3]Department of Computer Science, University of Sargodha, Pakistan
[4]Department of Allied Health Sciences, Superior University Lahore, Pakistan
Correspondence: shahidameer.khan@gmail.com

Abstract:

In today's digital world, users mainly rely on online platforms for various activities like shopping, social networking and streaming media. Therefore, recommender systems play crucial role in modern digital platforms as it guides the users by suggesting them content and items relevant to their interest. In this research study, we aim to develop an effective recommender system that accurately predicts unknown item ratings and make recommendations according to user's interest more precisely by applying collaborating filtering (cf) based techniques on user-item interaction. We applied various cf based techniques including user k-nearest neighbors and item k-nearest neighbors utilizing cosine similarity and pearson correlation and matrix factorization-based methods which are alternating least squares and singular value decomposition. Machine learning methods such as slope-one, baseline and co-clustering as well as deep learning models like neural collaborative filtering, recurrent neural networks and long short-term memory (lstm) networks. For the performance evaluation of these methods and models the metrics used are mean absolute error (mae) and root mean square error (rmse) on two datasets which are book-crossing and recipe reviews. The results indicate that lstm consistently outperforms rest of the applied algorithms by achieving the lowest rmse and mae values 1.06 and 0.31 on book-crossing and 0.44 and 0.37 on recipe-reviews dataset. The empirical analysis shows capability of lstm model to capture complex patterns and long-term dependencies in users' data. The findings highlight the effectiveness of deep learning models especially lstm in making accurate personalized recommendations greatly which can improve users' satisfaction and engagement.

Introduction

Evolution of social web have drastically changed the way users interact with online platforms due to this huge volume of user-generated content are produced on social media, e-commerce sites and other platforms. This huge volume of data available on online platforms challenging for users to find and select relevant content of their interest. Recommendation systems have become important tools for modern digital platforms addressing this challenge of information overload by suggesting content to users of their interest. These systems enhance user experience through making the recommendation according to user behavior and preferences *i.e.* Users rely on digital platforms for shopping, social networking and media streaming. Recommender systems make relevant and precise recommendations of products, connections and content that match their interest [1] [2]. Necessity of automated recommender system is very clear in this context to understand user's interest and make precisely personalized suggestions for them [3]. Authors [4] indicate benefits of using recommender system, amazon is really good at helping to find things to buy, and it makes them a lot of money about 35% of what amazon earns. Netflix is another example where recommendation systems shine. Mckinsey reports that 75% of what people watch on netflix is suggested by their clever algorithm [5] . In a paper by netflix executives, it's mentioned that this recommendation system actually saves netflix about $1 billion every year [6]. These systems have become super important for businesses too. They help companies keep you engaged with their platforms, which can lead to more sales and happier customers. And as technology keeps improving, recommendation systems are only going to get better at helping you find exactly what you are looking for online.

Though the availability of large amount of online information provides opportunities but this exponential increase in available data become a challenge for users to access relevant and meaningful information due to huge volume of information [7]. The users can discover relevant content across multiple platforms, including streaming services and e-commerce websites by using recommendation systems. To provide item or product suggestions these systems use different techniques to examine user behavior, preferences and item attributes. Cf one of the foundational techniques which identify similarities between users and items, suggesting items that similar users have enjoyed, compares user interactions and item ratings. With help of algorithms and methods based on similarity metrics, matrix factorization, machine and deep learning models we can smartly manage the working of recommender systems so it work efficiently and accurately as these techniques help in analyzing large datasets having complex patterns. Similarity metrics can help in finding the resemblance between the item and users to match user's interests. On the other hand, mf techniques break down the user-item interaction matrix to reveal latent features that capture fundamental relationships, even with sparse data enabling more accurate predictions. Ml and dl methods, for instance large datasets is formed by neural networks, excel at learning complex patterns that allow recommendation systems to apprehend intricate relationships in user-item interactions and provide more accurate suggestions. Recommendation systems boost user experience by providing relevant and appealing content personalized to individual preferences.
Recommender systems have become the fast evolving topics in the field of big data. To enhance the recommendation process various researches have been conducted highlighting its critical role in marketing and business. Increased orders and sales correlate with higher product rating typically because our decision to select product depends upon item or item rating or review and other people opinion. This relationship underscores the importance of accurate and effective recommendations.

In this proposed work, our aim is to apply collaborative filtering techniques to various datasets to predict unknown item ratings to address the data insufficiency problem, the accuracy improvement and item sparsity problem by estimation of unknown item rating using cf based algorithms. The existing studies mainly follow cosine similarity and matrix factorization based approaches and there is a research gap to apply and compare various machine learning and deep learning algorithms for finding the optimal results in recommender systems.

Objectives of the study is to apply various algorithms and models based on cf technique included similarity metrics like user k-nn, item k-nn using pearson correlation and cosine similarity, machine learning models based on mf technique such as mf (svd) and mf (als) and methods like slope-one, baseline and co-clustering. Various deep learning models like ncf, rnn and lstm to predict user-item ratings and offer personalized recommendations. To analyze the performance of methods and models used with the help of metrics like rmse and mae across two datasets: book-crossing and recipe reviews. To determine the top performing cf based methods by comparing similarity metrics, matrix factorization, machine and deep learning models analyzing their accuracy in predicting user and item ratings. To compare the results of the proposed methods with existing recommendation approaches to validate the effectiveness of the new techniques. The aim of our research is to achieve these objectives to boost the perfection and reliability to recommendation systems, contributing to their development and practical implementation across various realms
This research study achieves the following contributions:

- Application of cf technique using various algorithms and models included similarity metrics like user k-nn, item k-nn using pearson correlation and cosine similarity, machine learning models based on mf technique such as mf (svd) and mf (als) and methods like slope-one, baseline and co-clustering. Deep learning models like ncf, rnn and lstm to predict user-item ratings and offer personalized recommendations
- Analysis of different algorithms shows that lstm outperformed other algorithms which resulted rmse and mae 1.06 and 0.31 on book-crossing and 0.44 and 0.37 on recipe-reviews dataset.
- The results of proposed lstm model better than existing recommendation approaches, shown its effectiveness through performance evaluation measures.

Related work

In order to infer user preferences and make accurate predictions, the several recommendation techniques are used. The purpose of this review is to critically analyze the existing studies on the role of recommender system for the e-commerce. In the recent years, the considerable attention forwards towards the understanding of these system to take benefit from them. The studies reviewed various online product recommendation techniques based on collaborating filtering, content-based filtering and hybrid models.

Collaborative filtering

Collaborative filtering is a technique used in recommendation systems to forecast a user's choices or interests and enhance their satisfaction by examining the opinions and conduct of a group of users. Rather than relying on explicit information about items or users, for instance item descriptions or user profiles, the technique of collaborative filtering filters information and make predictions from the past interaction of a user and items to recommend similar content material

Following study [8] addresses the challenge of explainable recommendations in systems by integrating collaborative filtering with knowledge base graph (cfkg). The solution involves constructing a user-item knowledge graph to embed heterogeneous entities and their relations, facilitating personalized recommendations. The soft matching algorithm is a unique feature which produces explanations for suggested items, enhancing the models explain ability.

A new model linear residual graph convolutional collaborative filtering (lr-gccf) is introduced in another study [9] encompassing linear embedding propagation and residual learning to address the over-smoothing issue common in deep gcns. Better performance and scalability achieved on real-world datasets by simplifying model structure and concentrating linear transformations.

Social media connected the whole world across online platforms creating a virtual society and generating huge online information. Study [10-31] focused on recommender systems which do work of filtering information for users and suggest those items of their taste. With the use of cf different methods for predicting the user ratings were compared. The study found that item k-nn using pearson correlation was the most effective than other methods used.

Utilization of graph neural networks for collaborating filtering (gnn-based cf) as recommendation technique for making certain items more evident in recommendation systems used by websites [11]. This method smartly manages the item connections and primary network of user in contrast to previous approach that might use fake accounts or manipulate data in less direct ways. The careful selection of connections to emphasize, it can make a product appear more appealing and more possibility to be suggested, without needing to resort to dishonest tactics.

The user privacy is protected by the differential privacy (dp) in collaborative filtering recommendation systems [12]. Though, trade-off between privacy and fairness is highlighted, where biased recommendations led by dp-sgd is favoring active users over inactive ones. To acknowledge this, the research proposes dp-fair, a combining framework of dp and fairness constraints. In order to ensure fairness, dp-fair carried out private training using dp-sgd and then re-visit recommendations to make fairness between active and inactive users.

due to the lack of ratings for many items commonly in e-commerce domains the sparsity in collaborative filtering (cf) recommender systems is really challenging [13]. To overcome this problem, the studies recommend an approach combining clustering (k-means) and dimension reduction (svd). The results demonstrate that in comparison to previous studies, their method significantly reduces the rmse value, indicating the effectiveness of combining clustering and dimension reduction to tackle sparsity problems in cf recommender systems.
A study proposed new method in which by grouping users with similar interests, online shopping recommendations are improved [14-32]. This method ordered clustering based algorithm (oca) which helps overcome common challenges like recommending new items (cold-start problem) or dealing with limited user data (data sparsity) by analyzing users' past behaviors and preferences.
The sentiment analysis integrated with collaborative filtering (sa based bi-lstm) methods to enhance the precision and customization of suggestions in online stores [15-33]. The projected approach for sentiment analysis exploits a bi-lstm model, which captures background dependencies and information in sequential data, increasing the accuracy of sentiment analysis. Combination of sentiment analysis and collaborative filtering outcomes in a novel recommendation system that outclasses traditional methods for prediction accuracy.
To make recommender systems better especially the kind collaborating filtering a big issue with such type of systems is that they miss some user ratings which can cause of less accurate suggestions. To fix it the study [16] suggested to fill in the missing ratings values first then using simple methods such as averaging the available ratings before making recommendations. Evaluation made with data from amazon reviews and this approach made the recommendations more accurate suggested that filling in missing data can help make these systems work better.

Table 1: collaborating filtering

Ref	Year	Model name	Dataset	Results (% / raw-count)
[8]	2018	Cfkg	Amazon (cds and vinyl)	Ndcg: 5.56, recall: 7.94, ht: 17.55, pre: 2.19
[9]	2020	Lr-gccf	Amazon books, gowalla	Hr@n (0.02 - 0.03) , ndcg@n (0.11 - 0.23)
[10]	2022	Item k-nn(pearson correlation)	Ml 100k, ml 1m, ciao dvd	Rmse: 0.933, 0.879, 0.964 mae: 0.734, 0.690, 0.734
[11]	2023	Gnn-based cf	Gowalla, yelp2018	Phn@50: 23.4,23.3, 20.5 & 23.1, 16.7, 13.0
[12]	2023	Dp-fair	Amazon datasets	Ndcg: 11.2-13.5 , f1: 4.10-5.36
[13]	2023	Svd & k-means clustering	Amazon review	Rmse reduction: 7.270
[14]	2023	Oca	Amazon review data 2018	Precision:11.03, recall: 33.58, measure: 22.86
[15]	2023	Sa based bi-lstm	Kindle book, amazon music.	Accuracy: 93, 94, f1-score: 94,96, auc: 92, 78
[16]	2024	Cf with mean imputation	Amazon customer review	Rmse: 0.3198, mae: 0.1316

Content based filtering

Content-based filtering is a recommendation system that suggests things to someone by paying attention to what someone liked before. It looks at the details of items, like genres or themes, and then recommends stuff that's similar to what someone shown interest in. So, if someone into action movies, it might suggest more action-packed films so it focuses on personal preferences. However, it might not surprise with totally new things outside someone's usual interests.

A sequential hierarchical attention network (shan) to address the next item recommendation challenge by leveraging user's sequential behavior [17]. It innovatively used a hierarchical attention mechanism to dynamically integrate both long and short term preferences of users, distinguishing it from existing approaches that might not adequately capture the evolving nature of user interests. Shan achieves superior performance in predicting user preferences, as demonstrated by its significant improvements over existing models regarding recall and auc on real-world datasets.

New next-item suggestion framework using sequential hypergraphs. In these recommendation framework, the hypergraphs are employed to demonstrate multiple convolutional layers and short-term item correlations are utilized to find multi-order connections within the hypergraph [18]. It draw relationship between and incorporates short-term user intent and dynamic item embedding using a fusion layer before feeding the data into a self-attention layer for dynamic user modeling. In this paper, the challenge of capturing dynamic user preferences in sequential recommendation scenarios are suggested to be addressed in this novel approach.

A system content-based recommender system is designed to suggest items on e-commerce platforms in more efficient way [21-34]. The content-oriented filtering is used, particularly analyzing product documents are focused with tf-idf and vector space models. This tactic aids solving cold start problem, where making recommendations are preferred over user interests. The system exhibited profound results in testing, representing that on the bases of content similarity, relevant products can be accurately suggested to users based on textual data using various textual processing techniques.

This paper [22-36] introduces a specialized image retrieval system for the carpet e-commerce sector, leveraging advancements in content-based image retrieval techniques. Two novel methods quantized color layout descriptor (qcld) and dominant color descriptor with improved partitioning (dcdip) are proposed which used histogram, aimed at improving search performance and accuracy by focusing on color layout and dominant color descriptors, respectively.

The ebay users are recommended with items by learning how to illustrate both items and users in such manners they can be easily complemented by innovative system using cluster-based knn algorithm [23]. The characteristics of items and the diverse activities of users on ebay are focused to tackles the task of recommending new items (cold-start problem). The significant improvement in relevance of recommendations system is shown by system compared to older methods, making it easier for users to find their interested items.

Some studies have looked at suggesting the most popular items, but that doesn't always make the most money for the company. Study [24-35] introduces a method called profit-support fuzzy association rule mining (p-farm) it uses values between 0 and 1. It looks at both popular items and how much money they make for the company. P-farm helps companies decide which products to suggest to customers, making more profit with fewer rules.

A movie recommendation tool was developed by using content-based filtering and different semantic techniques for the enhancement of recommendation accuracy [19-37]. Research compared various methods and techniques for processing movie reviews including advanced word embedding models which are bert, gpt-2, roberta and other is term frequency-inverse document frequency (tf-idf). In experimental work the methods used roberta outperformed other methods by delivering the best result of accurate predictions with high precision and accuracy. Research shown that while tf-idf produced higher errors semantic and other method which is roberta utilize deep learning and transformer-based models which improved the system's performance.

Table 2: content based filtering

Ref	Year	Model name	Dataset	Results (% / raw-count)
[17]	2018	Shan	Tmall, gowalla	Recall@20: 33.6 , 9.8
[18]	2020	Sequential hypergraph recommendation	Amazon, etsy, goodreads	Hit@1:0.12, 0.47, 0.28 , ndcg@1:0.12, 0.28, 0.47 hit@5:0.32, 0.70, 0.62 , mrr:0.23, 0.57, 0.44
[20]	2020	Content-based filtering system	E-commerce product data	Average recall: 0.84, average precision:0.78
[21]	2021	Qcld , dcdip	Persian carpet images	Anmrr: 0.216, retrieval time: 2.84sec, anmrr: 0.125, retrieval time: 8.15sec
[22]	2021	Personalized embedding	Ebay e-commerce data	Recall@k (k=1-40) : 50, 8.3, 12.5
[23]	2023	P-farm	E-commerce transaction	Total profit: 39 , average profit: 19.5
[19]	2024	Roberta	Multi-platforms (movies data)	Accuracy: 95.59, precision: 95.76, loss: 0.6514, recall: 95.41, f1-score: 95.58

Hybrid models

Hybrid recommender systems are like super-smart suggestion machines. They mix and match different ways of giving recommendations to make sure they're really good and cover a lot of different things. There are two main types: one looks at what things are about (content-based), and the other checks out what similar people like (collaborative-based). By combining these, they make sure someone get awesome and varied suggestions that suit his tastes.

An innovative approach for mobile marketing recommendation, termed as user location based via convolutional neural network (lbcnn) model [25]. In this novel approach, the users' location-based behaviors within various time frames are taken into account. It extracted the features to model preferences from different dimensions. By training classifier using cnn, the model shows the efficacy of incorporating spatial-temporal user behavior data for personalized recommendations.

A study presents a new approach for sequential recommendation in model ma-gnn (memory augmented graph neural network) [26]. Which efficiently apprehend both the short-term and long-term user preferences.

Recommendation accuracy is improved by this modeling item co-occurrence patterns and employing a unique user interest fusion mechanism. The existing models convincingly outperformed by this approach across various datasets, revealing its effectiveness in addressing the diverse nature of user interests and improving recommendation systems.

To boost the training speed and accuracy of recommender systems, introduction of fedfast [27] a federated learning framework designed is cited. In this method, fedfast significantly lessens communication rounds and training efforts by exploiting innovative sampling and aggregation techniques in comparison to traditional federated learning methods. The experiment displays that fedfast is more effective than baseline federated averaging method (fedavg).

A hybrid approach to product recommendation systems [28], which combines machine learning association and clustering algorithms. The aim is to provide both demographic representation of consumer profiles connected to these associations and product associations. By integrating association rule mining and clustering algorithms, the system is able to identify detailed product associations and the demographic profiles of customers associated with them.

The accuracy and effectiveness of product recommendations are improved by hybrid recommendation systems in which different recommendation techniques are combined [29]. Three new algorithms are introduced *i.e.,* tf-idf vectorization, cosine similarity and svd. User data and item similarities are analyzed to predict item ratings by using these algorithms.

A recommendation system for fashion retail shops that addresses various challenges such as cold start problems, computational complexity, low returns in physical stores and seasonal product effects [30]. The system employs a multi clustering approach based on mining techniques to predict customer purchase behavior and provide personalized recommendations. The novelty lies in the utilization of clustering methods considering rfm values and multiple seasons of products, along with enriching customer behavior data.

Another study [24] presents a movie recommendation tool rmsprop combines different techniques. This system first used cf to make initial recommendations based on user interactions. The made recommendations are then further refined using content-based filtering which focuses on the content of movies and user preferences. And final step is involvement of cnn to classify and enhance recommendations further. Then further system processed data from movie titles and user reviews including tweets about films. This approach effectively integrates different filtering methods and deep learning to make more accurate movie suggestions.

Table 3: hybrid model

Ref	Year	Model name	Dataset	Results (% / raw-count)
[25]	2019	Lbcnn	Alibaba mobile recommendation	Accuracy: 80, recall: 8.14, f1-score: 8.07
[26]	2020	Ma-gnn	Real-world datasets	Recall@10: 4.42 - 12.36, ndcg@10: 2.14 - 12.72
[27]	2020	Fedfast	Tripadvisor, yelp, ml100k, ml1m	Hr@10: 0.50, 0.62, 0.89, 0.71; ndcg@10: 0.26, 0.36, 0.62, 0.44

[28]	2022	Hybrid ml-based recommender system	Hygiene product	Support:17.485, lift: 2.276, confidence:77.087, rule support: 13.478,
[29]	2022	hrs	E-commerce website data	Lift: 1.6018
[30]	2022	Multi clustering recommender system	Retail company tessilform,	Conversion rate 3.48
[24]	2024	Cnn model with rmsprop	Imdb and twitter film review	Mae 0.8643, rmse 0.6325 , accuracy 88.40

Recommender system help users to suggest items relevant to their interest. The above literature review reveal that several works done in ratings prediction but there is no such utilization of advance methods like advance deep learning models such as lstm to predict the item's ratings and their comparison with typical used techniques like matrix factorization and similarity based metrics. Higher the ratings directly lead to user satisfaction and ultimately retention of users and more sale. There is gap in effectively combining the rating prediction with personalized recommendations using deep models and comparing them with typical methods like similarity metrics and matrix factorization. There is also need to address algorithmic bias which can favor popular items and overlook niche interests. Covering these gaps involves developing approaches that integrate rating prediction with recommendation strategies while ensuring the enhancement of overall user experience.

Proposed work methodology

In the start of this process, removing inconsistencies are involved in dataset cleansing. By following data cleaning, data normalization is performed to confirm that the data is on a common scale without differences in the values ranges. Then, various algorithms are implemented to reveal patterns and provide personalized recommendations. Both collaborative filtering methods and advanced machine learning models are involved in these algorithms. Next, predictive accuracy of our models is evaluated by using suitable performance metrics. By using these metrics, it is permitted to measure accuracy of proposed recommendation system for predicting user preferences.

These steps are followed with the aim to have new insights into recommendation systems and enhance their effectiveness. The summary of the research methodology is illustrated in the diagram in fig. (2).

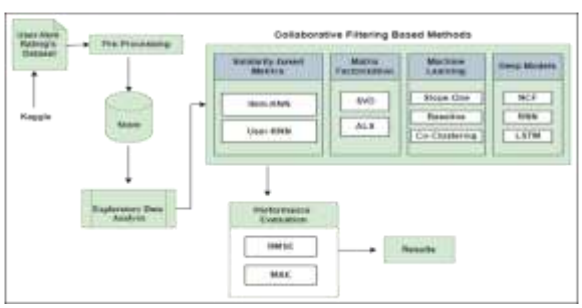

Figure 2: setup of research methodology.

DOI:10.48047/HM.V10.I2.2024.2241-2260

In this study, a popular platform for sharing datasets, kaggle provided the used user-item ratings dataset. These data set include book-crossing user review ratings[1] and recipe reviews and user feedback dataset[2]. The format of these datasets was structured in a csv (comma-separated values), which in turn provide easy access and manipulation during analysis. By using this format, efficient data handling is facilitated, that allows seamless exploration and processing of the dataset's contents.

Pre-processing

This phase involves removing null values or filling missing values for enhancement of data accuracy, completeness, and suitability for subsequent analysis tasks. Particularly, by implementing normalization operations to the dataset, its format and scale is standardized and more effective analysis and model building facilitated. We undertake these preprocessing steps systematically for ensuring integrity, reliability of data and in later stages of the research lay a solid foundation for robust and insightful analyses.

Application of collaborating filtering based methods

The context of this research includes rating prediction as a prime focus of the modeling process. The generation of predictions for user-item ratings on the bases of collaborative filtering algorithms, ml models, and dl techniques is involved in this process. The aim is the accurate estimation of how items are rated by users, they have not yet interacted with so that recommendation system allowed recommending personalized and relevant items. The evaluation of performance and effectiveness of the recommendation algorithms used in the study is achieved using predicted ratings as a key metric. The assessment of the accuracy and reliability of the rating predictions allows us to measure the overall quality and ability of recommendation system to provide valuable suggestions to users.

Pearson correlation

User or item knn with pearson correlation is a collaborative filtering algorithm. By finding similar users or items on the bases of the past ratings, this algorithm predicts the user's rating to an item. This method is very simple and effective in capturing user preferences on the bases of their similar rating behavior. It recognizes users or items with similar rating patterns and recommend items to the active user liked by similar users. In order to measure their similarity, at first pearson correlation coefficient is computed between pairs of users or items. Based on this correlation. Most similar users or items to active user or item are selected. Afterwards, by weighing and aggregating the ratings from these similar users or items, the not rated user rating for items is predicted.

The similarity in between users is calculated through (1) provides pearson correlation:

$$\boldsymbol{w_{u1,u2}} = \frac{\sum i \in I_{u1u2} (r_{u1i} - \bar{\bar{r}}_{\bar{u}\bar{1}})(r_{u2i} - \bar{\bar{r}}_{\bar{u}\bar{2}})}{\sqrt{\sum_{i \in I_{u1u2}} (r_{u1i} - \bar{\bar{r}}_{\bar{u}\bar{1}})^2} \sqrt{\sum_{i \in I_{u1u2}} (r_{u2i} - \bar{\bar{r}}_{\bar{u}\bar{2}})^2}} \qquad (1)$$

The similarity in between items is calculated through (2) provides pearson correlation:

$$\boldsymbol{w_{i1,i2}} = \frac{\sum u \in U_{i1i2} (r_{ui1} - \bar{\bar{r}}_{\bar{\imath}\bar{1}})(r_{ui2} - \bar{\bar{r}}_{\bar{\imath}\bar{2}})}{\sqrt{\sum_{u \in U_{i1i2}} (r_{ui1} - \bar{\bar{r}}_{\bar{\imath}\bar{1}})^2} \sqrt{\sum_{u \in U_{i1i2}} (r_{ui2} - \bar{\bar{r}}_{\bar{\imath}\bar{2}})^2}} \qquad (2)$$

Where $\boldsymbol{w_{u1,u2}}$ is the weight of similarity between users' $\boldsymbol{u1}$ and $\boldsymbol{u2}$. $\boldsymbol{w_{i1,i2}}$ is the weight of similarity between items' $\boldsymbol{i1}$ and $\boldsymbol{i2}$. $\boldsymbol{r_{ui}}$ is ratings of user $\boldsymbol{u}$ for item $\boldsymbol{i}$. $\bar{\boldsymbol{r}}_{\imath}$ and $\bar{\bar{\boldsymbol{r}}}_{\bar{u}}$ are average ratings of items and users.

Cosine-similarity

By finding similar items or users on the bases of ratings, a collaborative filtering algorithm, item or user k-nn with cosine similarity is applied to predicting user's rating to item. The item or user similarities based on their rating vectors across users or items is captured by selecting this approach. It assists to recommend items related to those liked by user in past, leveraging item-item or user-user relationships. The measurement of cosine similarity is achieved between pairs of items or users using their rating vectors. Those items are selected that most similar to those the active users highly rated. Active user's rating for unrated items is predicted by aggregating and weighing the ratings of these similar items or users.

To compute these similarities, (3) is applied. This offers the measurement of resemblance between users based on their respective vectors.

$$\mathrm{sim}(\vec{u}1, \vec{u}2) = \frac{\sum i \in I_{u1u2} \; r_{u1i} \cdot r_{u2i}}{\sqrt{\sum_{i\in I_{u1u2}} (r_{u1i})^2} \times \sqrt{\sum_{i\in I_{u1u2}} (r_{u2i})^2}} \qquad (3)$$

(4) is applied to measure resemblance between items based on their respective vectors.

$$\mathrm{sim}(\vec{i}1, \vec{i}2) = \frac{\sum u \in U_{i1i2} \; r_{ui1} \cdot r_{ui2}}{\sqrt{\sum_{u\in U_{i1i2}} (r_{ui1})^2} \times \sqrt{\sum_{u\in U_{i1i2}} (r_{ui2})^2}} \qquad (4)$$

Where $\mathbf{r_{ui}}$ is ratings of user $\mathbf{u}$ for item $\mathbf{i}$. $\mathbf{U_{i1i2}}$ set of users who rated items $\mathbf{i1}$ and $\mathbf{i2}$. $\mathbf{I_{u1u2}}$ set of items rated by users $\mathbf{u1}$ and $\mathbf{u2}$.

Machine learning models

Ml models enhance the cf by improving the accuracy and personalization of recommendations. The advantage they provide it the handling of data sparsity common issue in cf where user only rated few items. They have ability to handle complex and non-linear relationship between users and items and they continue to learn from new user interaction to make sure recommendation keep relevant according to user preferences.

Slope-one

Slope-one is used for predicting user-item ratings. This technique is a simple yet effective collaborative filtering algorithm. In it, the predicted rating of item for the user is computed on the bases of the differences between ratings for items already have rated by user and the average differences between these ratings and the ratings of other items. Using slope-one, the rating of item j for user u based on user's rating of another item i and difference of average rating between items i and j across all users are predicted. The average deviation of item j ratings from item i ratings is calculated by this algorithm and this deviation is used in adjustment of the user's rating for item i in order to predict the rating for the item j.

$$\boldsymbol{P(u,i)} = \frac{\mathbf{1}}{|\boldsymbol{I_i}|} \boldsymbol{\Sigma\, j \in I_i \,(r_{u,j} + dev(i,j))} \qquad \mathbf{(5)}$$

Where $\boldsymbol{P(u,i)}$ predicted rating of user u on the item i. $\boldsymbol{I_i}$ set of items both user $\boldsymbol{u}$ and other users have rated. $\boldsymbol{r_{u,j}}$ rating of user $\boldsymbol{u}$ on the item $\boldsymbol{j}$. $\boldsymbol{dev(i,j)}$ average deviation between ratings of items $\boldsymbol{i}$ and $\boldsymbol{j}$.

Baseline

Baseline algorithms is a simple but effective method for cf making rating predictions. It predict user's ratings by relying on basic statistical measures instead of using complex models. In order to get quick and almost accurate prediction of user preferences which is primary goal of baseline algorithms. The three main components are global average rating, user bias and item bias in baseline prediction algorithm in which global average rating is the mean of all ratings in the dataset. The direction of users to rate items lower or higher to global average is reflected by user bias and items to receive lower or higher ratings than the global average is accounted by item bias.

The expression of prediction formula for a baseline algorithm is given as:

$$\boldsymbol{P(u, i) = \mu + b_u + b_i} \qquad (6)$$

Where $\boldsymbol{P(u, i)}$ predicted rating of user u on the item i, μ is global average rating, b_u bias of user u (i.e., how much lower or higher to average this user rates items) and b_i bias for item i (i.e., how much lower or higher to average this item is rated).

3.2.3.1 Co-clustering

Co-clustering technique is also called bi-clustering. This approach clusters rows and columns of a data matrix simultaneously. By grouping similar items and users together, this method is particularly valuable for discovering latent structures and patterns in the data. Co-clustering improves the accuracy of recommendations by recognizing clusters of users with similar preferences and clusters of items that receive similar ratings. The work of co-clustering is to reduce the error in predicting ratings by iteratively updating user and item clusters. The two main steps are involved in this process *i.e.,* assigning users to user clusters and items to item clusters, and then these clusters are refined in order to enhance the accuracy of the rating predictions. This repetitive process goes on until achievement of convergence.

$$\boldsymbol{\hat{r}_{u,i} = \overline{C_{ui}} + (\mu_u - \overline{C_u}) + (\mu_i - \overline{C_i})} \qquad \textbf{(7)}$$

Where $\hat{r}_{u,i}$ predicted rating of user u and item i, $\overline{C_{ui}}$ average rating of the user-item cluster C_{ui}, μ_u average rating given by user u, $\overline{C_u}$ average rating of the user cluster u, μ_i average rating received by item i and $\overline{C_i}$ average rating for the item cluster i

Matrix factorization techniques

By using these techniques, the user-item interaction matrix $\boldsymbol{R}$ is decomposed into lower-dimensional matrices so that the users and items are effectively represented in a latent feature space. The objective of decomposed matrices is to reconstruct the original matrix $\boldsymbol{R}$ with possible accuracy.

Singular value decomposition (svd)

In this technique, user and item interaction matrix is decomposed to lower dimensional matrices so as to obtain latent features that explain observed ratings. The one such technique is svd. The matrix is decomposed into orthogonal matrices of singular values in svd. The dimensionality of the matrix is reduced to expose hidden patterns and improves recommendation accuracy.

In svd the item and user matrix $\boldsymbol{R}$ is decomposed into three matrices $\boldsymbol{U}$, σ and $\boldsymbol{V^T}$. These matrices represent users, latent factors, and items, respectively. The product of these matrices provides approximation of $\boldsymbol{R}$:

$$\mathbf{R} \approx \mathbf{U}\, \boldsymbol{\Sigma}\, \boldsymbol{V^T} \qquad (8)$$

The σ is diagonal matrix for singular values while U and V are orthogonal matrices. For item i and user u, the predicted rating $\hat{r}_{u,i}$ is given by using svd:

$$\boldsymbol{\hat{r}_{u,i} = \bar{r}_u + (u_u^T\, v_i)} \qquad (9)$$

Where mean rating $r_{\bar{u}}$ for user u, u_u is latent factor vector of user u and v_i is the latent factor vector for item i.

Alternating least squares (als)

Alternating least squares (als) is a type of matrix factorization technique. This approach is exploited to predict user-item interactions in collaborative filtering by alternatingly solving least squares optimization problems, the item and user interaction matrix is decomposed to lower dimensional latent factor matrices. Through alternating optimization, als suggests a scalable solution by efficiently computing factor matrices for large-scale recommendation systems.

The alternation of it is done to minimize the reconstruction error in which one matrix is fixed (either user matrix $\boldsymbol{U}$ or item matrix $\boldsymbol{V}$) and other is optimized. This process continues iteratively until convergence, resulting in matrices $\boldsymbol{U}$ and $\boldsymbol{V}$ that approximate original matrix $\boldsymbol{R}$.

The function aims to minimize regularized squared error in between the observed and predicted ratings:

$$\boldsymbol{min_{U,V}\, \Sigma_{(u,i)} \in R(r_{u,i} - u_u^T v_i)^2 + \lambda\, (\,||u_u||^2 + ||v_i||^2)} \qquad \textbf{(10)}$$

Where mean rating $r_{\bar{u}}$ for user u, u_u is latent factor vector of user u and v_i is the latent factor vector for item i.

Deep learning models

The complex, non-linear interactions between items and users are captured by dl models. These models are effective in handling the sparse data and non-linear relationships as they have ability to learn intricate and high dimensional user and item representation in cf technique as they have ability to extract features from raw interaction data. Their ability to handle large scale data and adopt diverse user behavior makes them valuable tool in enhancing the effectiveness and robustness of cf methods.

Neural collaborative filtering (ncf)

This is an advanced recommendation technique that aim of combining neural networks with collaborative filtering to learn item and user interactions directly from data, leveraging both explicit and implicit feedback. In ncf, by learning embedding's for users and items, latent factors can be captured, influencing user preferences more effectively as compared to traditional methods. To learn user and item embedding, a multi-layer neural network architecture is typically employed in ncf. Furthermore, in order to optimize embedding and minimizing the prediction error, the non-linear activation functions and regularization techniques are used.

Recurrent neural networks (rnn)

Rnn are specialized neural network which handle sequential data and make them compatible for tasks having time series or sequential dependencies and sound like sequential recommendation systems. Internal memory cells in rnns which help to maintain information about previous inputs. Due to its memory capability rnns are able to capture temporal dependencies in sequential data which is important in order to comprehending the user behavior over time in recommendation systems. In rnns sequential data is processed through iteration of one element at a time through the input sequences and a hidden state is maintained to encapsulate information from previous inputs.

Long short-term memory networks (lstm)

Lstm fall under the category of rnn and its architecture designed to handle the limitations presence in traditional rnns to identify and handling of long-term relationship in sequential data. Lstm networks control the information flow over the network with help of a memory cell and numerous gating mechanisms. Lstms is able to make that user preferences and behavior over time in recommendation systems is modeled by capturing long range of dependencies and maintaining the context over extended sequences. In addition to the memory cell the output gate, forget gate and input gate are three main elements of lstm. These components make the flow of information through network by working together.

Experimental setup:

The below section involves investigation of crucial information regarding the dataset utilized. The practicality of our model is analyzed by employing performance evaluation metrics. For this research the experimental work performed on two real world datasets taken from kaggle a well-known dataset repository.
Dataset

The book crossing dataset having the anonymized data only with demographic information from 1,149,780 ratings are provided by 278,858 users overall (both explicit and implicit) about 271,379 books.

The "recipe reviews and user feedback dataset" consists of a comprehensive collection of data which encircle different facets of recipe reviews and user interactions. The essential details are included in this data for example, recipe names, unique recipe codes, their rankings among top 100 recipes list, and user information such as user ids, usernames, and internal user reputation scores. To compute users' thoughts towards recipes, 1 to 5 star rating scale is used, and the absence of a rating indicates a score of 0.
Performance evaluation measures:

It is the criteria to measure the efficiency and utility of a model. This metrics used to assess the level of performance of the designated task. To scale performance of this model, several metrics are available. By comprehending the calculations of these metrics, the most appropriate model can be selected on the bases of quantitative measures within those criteria.

Sr. No	Metrics	Equation	Description
1	Root mean square error (rmse)	$\boldsymbol{RMSE} = \sqrt{\frac{\sum_{i=1}^{m}(\boldsymbol{q}-\boldsymbol{r})^2}{\boldsymbol{m}}}$	Rmse aids us to measure the difference of our predictions from the actual values. 'Q' is our predicted value, while 'r' is true known value. The accuracy of our predictions can be figured out by using this equation overall.
2	Mean absolute error (mae)	$\text{Mae} = \frac{1}{\text{m}}\sum_{\text{I}=1}^{\text{m}} \lvert\text{yi}-\text{x}\rvert$	Difference between actual and forecasted values is measured in this method. Where 'm' is the quantity of errors in numbers, while 'y' represents the prediction and 'x' is true value. By using this formula, on average, we can measure the extent of deviation of our predictions from the real values.

Results and discussion

The performance evaluation of all algorithm is measured with the help of metrics rmse and mae and then identification of best performing algorithm from our evaluations by comparing them and managing to provide visualizations of their relative strengths.

HistoryofMedicine,2024,10(2):2241-2260
DOI:10.48047/HM.V10.I2.2024.2241-2260

The purpose of this section in this research is to make a comprehensive analysis of our experimental results and making the understanding of the performance and effectiveness of various recommendation methods across datasets used.

Comparison of machine learning with deep models

In this comparison advanced deep learning models such as rnn, ncf and lstm are compared with machine learning algorithms such as slope-one, co-clustering and baseline methods shown in the table 4. The purpose is to check whether the deep learning models have better capability to capture the users and items complex and non-linear interactions.

Performance comparison of these models shown that on both datasets book-crossing and recipe reviews lstm have achieved the lowest rmse and mae values and outperformed machine learning algorithms as well as both ncf and rnn. The result highlights the strength of lstm visualized in fig. 3 for modeling user preferences and making accurate predictions. However, a remarkable competitiveness is shown by co-clustering by achieving lower rmse and mae values compared to ncf and rnn but due to the robust performance and advantage by lstm in accuracy predictions co-clustering did not compete lstm. Over all, in this assessment high accuracy is offered by deep learning models, particularly lstm as compared to other methods.

Table 4: comparison of machine learning with deep models.

Algorithms	Rmse Book-crossing	Rmse Recipe reviews	Mae Book-crossing	Mae Recipe reviews
Slope-one	3.22	1.04	2.50	0.56
Baseline	3.27	1.18	2.41	0.66
Co-clustering	1.22	1.29	0.36	0.68
Ncf	3.51	0.86	2.49	0.53
Rnn	3.34	0.90	2.44	0.44
Lstm	**1.06**	**0.44**	**0.31**	**0.37**

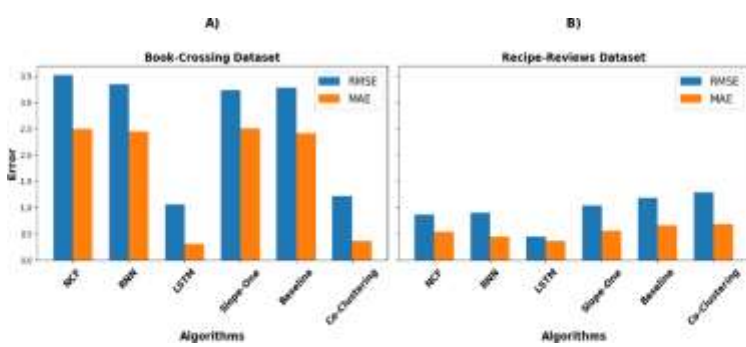

Figure 1: comparison of machine learning with deep models.

Comparison of lstm with matrix-based methods

Following the preliminary comparison, matrix factorization techniques such as mf (svd) and mf (als) compared with lstm in table 5. In our results, it is suggested that lstm still upheld superior performance despite mf (als) and mf (svd) are shown as powerful methods in the past. The lower rmse and mae values achieved by lstm in comparison to both mf (als) and mf (svd) on the book-crossing and recipe reviews datasets as shown in the fig. 4. This emphasized its efficiency to capture and leverage accurate predictions despite the complex pattern to handle in user-item interaction.

Table 5: comparison of lstm with matrix-based methods.

	Rmse		Mae	
Algorithms	**Book-crossing**	**Recipe reviews**	**Book-crossing**	**Recipe reviews**
Mf(svd)	4.45	1.89	2.20	5.76
Mf(als)	3.68	1.54	2.25	0.65
Lstm	**1.06**	**0.44**	**0.31**	**0.37**

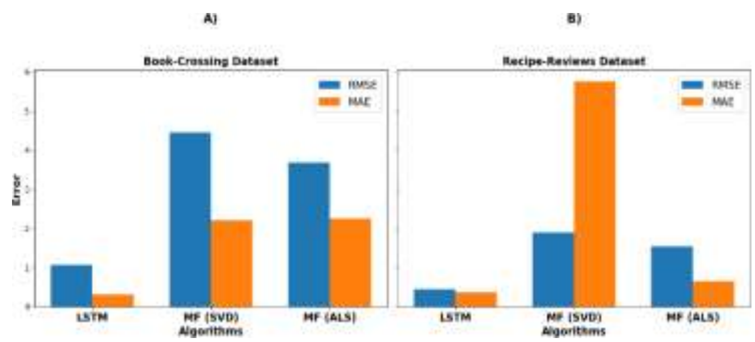

Figure 4: comparison of lstm with matrix-based methods.

Comparison lstm with similarity-based metrics

In this section an inclusive comparison is performed between lstm with user-user and item-item based cf. The item k-nn and user k-nn algorithms are evaluated against lstm and their performance is measured by using evaluation metrics rmse and mae shown in table 6. These results shown lstm's effectiveness mainly in a condition where understanding sequential patterns and long-term dependencies are important. Despite the computational cost required by lstm when it is compared to similarity-based metrics but its ability of modeling complex interactions can help in more accurate and personalized recommendations. The results of these comparison show lstm as a better option for recommendation tasks as it have strength of managing dynamic user preferences and item interactions.

Table 6: comparison of lstm with similarity-based metrics.

	Rmse		Mae	
Algorithms	**Book-crossing**	**Recipe reviews**	**Book-crossing**	**Recipe reviews**

Item k-nn (cosine similarity)	2.81	0.70	2.08	0.38
Item k-nn (pearson correlation)	2.70	0.70	1.84	0.37
User k-nn (cosine similarity)	3.16	0.78	2.60	0.43
User k-nn (pearson correlation)	3.40	0.82	2.81	0.45
Lstm	**1.06**	**0.44**	**0.31**	**0.37**

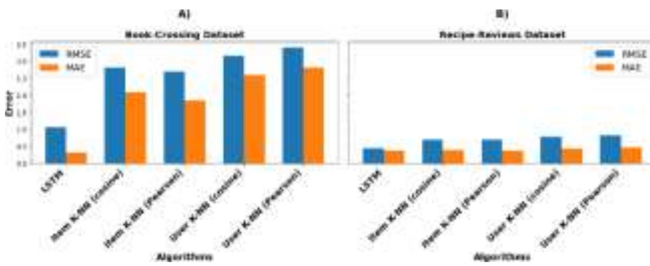

Figure 2: comparison of lstm with similarity-based metrics.

Optimal results comparison of all algorithms with proposed model

In this section a detailed comparison of all algorithms included in this research is conducted on the book-crossing and recipe reviews datasets in table 7. Our results suggested the lstm as dominate performer amongst all algorithms. It outclassed all other algorithms in terms of both rmse and mae across both datasets as visualized in fig. 6 the deep learning models, particularly lstm are proved as effective models in recommendation tasks. Lstm is proved to be a convincing choice for recommendation systems due to its ability to manage sequential data and recognize intricate patterns. Inclusively, our results approved proposed model lstm as a dominant network in recommendation accuracy resulted rmse and mae 1.06 and 0.31 on book-crossing and 0.44 and 0.37 on recipe-reviews dataset.

Table 7: comprehensive comparison of all methods.

	Rmse		**Mae**	
Algorithms	**Book-crossing**	**Recipe reviews**	**Book-crossing**	**Recipe reviews**
Item k-nn (cosine similarity)	2.81	0.70	2.08	0.38
Item k-nn (pearson correlation)	2.70	0.70	1.84	0.37
User k-nn (cosine similarity)	3.16	0.78	2.60	0.43
User k-nn (pearson correlation)	3.40	0.82	2.81	0.45

Mf(svd)	4.45	1.89	2.20	5.76
Mf(als)	3.68	1.54	2.25	0.65
Slope-one	3.22	1.04	2.50	0.56
Baseline	3.27	1.18	2.41	0.66
Co-clustering	1.22	1.29	0.36	0.68
Ncf	3.51	0.86	2.49	0.53
Rnn	3.34	0.90	2.44	0.44
Lstm (proposed)	**1.06**	**0.44**	**0.31**	**0.37**

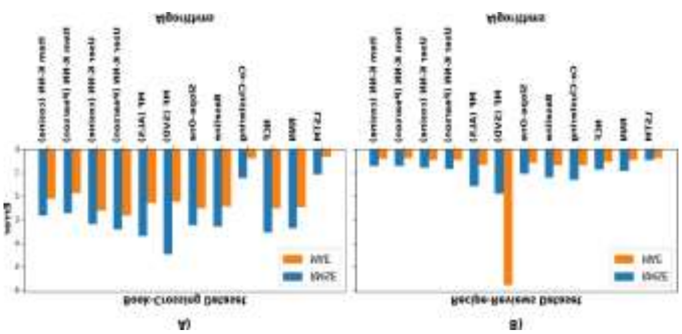

Figure 3: comprehensive comparison of all methods

Conclusion

This study aimed to explore various recommendation methods and models based on collaborating filtering techniques to make unknown item's rating prediction and making recommendations. Recommender system are tools which suggest personalized content to enhance user satisfaction and makes user interaction interesting and worthy. They play crucial role in current digital world having bundles of content as they help user to spend less amount of time to find content of their interest quickly. This study involved analyzing various methods and models using cf techniques to predict user and item ratings which are user k-nn, item k-nn using pearson correlation and cosine similarity, and matrix factorization techniques like mf (svd) and mf (als). We also applied machine learning methods such as slope-one, baseline methods and co-clustering as well as deep learning models like ncf, rnn and lstm networks.

In first step, we compared deep learning models including ncf, rnn and lstm networks with ml algorithms like slope-one, baseline and co-clustering methods. Then best identified methods from this comparison compared with matrix-based methods which are matrix factorization techniques such as mf (svd) and mf (als). Furthermore, similarity-based metrics like user-user cf algorithms such as user k-nn and item-item cf algorithms like item k-nn are evaluated. To do the performance evaluation of these methods we used metrics which are mae and rmse.

Our results shown that our proposed model lstm performed much better than other methods and algorithms across both datasets in terms of mae and rmse. Lstm resulted rmse and mae 1.06 and 0.31 on book-crossing and 0.44 and 0.37 on recipe-reviews dataset. Results indicated that capability of lstm is far better than other deep models to handle complex and capture the intricate patterns and long-term dependencies in the data.

Co-clustering and item-based collaborative filtering with pearson correlation also proved effective in peculiar perspectives, they generally fell short compared to more advanced methods. Our results shown the effectiveness of deep learning models is on top, especially lstm, on these datasets which clearly shown that its ability to capture data sparsity and handling the complex relations between user and item interaction. Building on this work, future efforts could combine collaborative filtering with context-aware methods to improve adaptability. Exploring transformer architectures may provide deeper insights into sequential patterns and rating prediction.

References

[1] y. Wang, w. Ma, m. Zhang, y. Liu, and s. Ma, "a survey on the fairness of recommender systems," *acm trans. Inf. Syst.*, vol. 41, no. 3, pp. 1–43, jul. 2023, doi: 10.1145/3547333.

[2] Bilal a, Fatima U. Diagnosis and treatment of hepatitis-c in bhalwal: case report. Palliative medicine and care international journal. 2021;4(1):1-3.

[3] J. Lu, d. Wu, m. Mao, w. Wang, and g. Zhang, "recommender system application developments: a survey," *decision support systems*, vol. 74, pp. 12–32, jun. 2015, doi: 10.1016/j.dss.2015.03.008.

[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, p. 141, jan. 2022, doi: 10.3390/electronics11010141.

[5] F. M. Groom and s. S. Jones, eds., *artificial intelligence and machine learning for business for non-engineers*, 1st ed. Crc press, 2019. Doi: 10.1201/9780367821654.

[6] A. Castrounis, *ai for people and business: a framework for better human experiences and business success*, first edition. Sebastopol, ca: o'reilly media, inc., 2019.

[7] Bilal a, ahmad s, tanvir f, tariq m, ramzan k, saleem m, saleem hg. Predictive modeling of n-acetyl transferase 2 single nucleotide polymorphisms and breast cancer risk using in-silco approaches. The journal of microbiology and molecular genetics. 2022 aug 31;3(2):105-21.

[8] C. A. Gomez-uribe and n. Hunt, "the netflix recommender system: algorithms, business value, and innovation," *acm trans. Manage. Inf. Syst.*, vol. 6, no. 4, pp. 1–19, jan. 2016, doi: 10.1145/2843948.

[9] M. M. Talha, h. U. Khan, s. Iqbal, m. Alghobiri, t. Iqbal, and m. Fayyaz, "deep learning in news recommender systems: a comprehensive survey, challenges and future trends," *neurocomputing*, vol. 562, p. 126881, dec. 2023, doi: 10.1016/j.neucom.2023.126881.

[10] Jawad m, bilal a, khan s, rizwan m, arshad m. Prevalence and awareness survey of tuberculosis in the suspected population of bajaur agency in fata, pakistan: prevalence and awareness survey of tuberculosis. Pakistan journal of health sciences. 2023 jun 30:56-61.

[11] Q. Ai, v. Azizi, x. Chen, and y. Zhang, "learning heterogeneous knowledge base embeddings for explainable recommendation," 2018, doi: 10.48550/arxiv.1805.03352.

[12] L. Chen, l. Wu, r. Hong, k. Zhang, and m. Wang, "revisiting graph based collaborative filtering: a linear residual graph convolutional network approach," 2020, doi: 10.48550/arxiv.2001.10167.

[13] S. Nudrat, h. U. Khan, s. Iqbal, m. M. Talha, f. K. Alarfaj, and n. Almusallam, "users' rating predictions using collaborating filtering based on users and items similarity measures," *computational intelligence and neuroscience*, vol. 2022, pp. 1–13, jul. 2022, doi: 10.1155/2022/2347641.

[14] Y. Wang, y. Liu, and z. Shen, "revisiting item promotion in gnn-based collaborative filtering: a masked targeted topological attack perspective," 2022, doi: 10.48550/arxiv.2208.09979.

[15] Z. Yang, y. Ge, c. Su, d. Wang, x. Zhao, and y. Ying, "fairness-aware differentially private collaborative filtering," 2023, doi: 10.48550/arxiv.2303.09527.

[16] D. B. T. Riyadi and z. K. A. Baizal, "collaborative filtering with dimension reduction technique and clustering for e-commerce product," vol. 7, 2023.

[17] Y. Gulzar, a. A. Alwan, r. M. Abdullah, a. Z. Abualkishik, and m. Oumrani, "oca: ordered clustering-based algorithm for e-commerce recommendation system," *sustainability*, vol. 15, no. 4, p. 2947, feb. 2023, doi: 10.3390/su15042947.

[18] Bilal a, nazar i, abbas a, shafiq hr, rafique a, saddiqa a, ali k, ullah mk. Existence, evolution, history and impact of ebola virus disease on humans: a mini review. International journal of environmental chemistry. 2024;10(1):1-5p.

[19] Umm-e-asma fs, shah ma, abbas kj, ramzan h, asif i, nija de, younas e, bilal a. Exploring the relationship between psychological stressors and myocardial infarctions in humans using statistical techniques. African journal biomedical research. 2024; 27(1)

[20] I. Karabila, n. Darraz, a. El-ansari, n. Alami, and m. El mallahi, "enhancing collaborative filtering-based recommender system using sentiment analysis," *future internet*, vol. 15, no. 7, p. 235, jul. 2023, doi: 10.3390/fi15070235.

[21] E. Hikmawati, h. Nugroho, and k. Surendro, "improve the quality of recommender systems based on collaborative filtering with missing data imputation," in *proceedings of the 2024 13th international conference on software and computer applications*, bali island indonesia: acm, feb. 2024, pp. 75–80. Doi: 10.1145/3651781.3651793.

[22] H. Ying *et al.*, "sequential recommender system based on hierarchical attention networks," in *proceedings of the twenty-seventh international joint conference on artificial intelligence*, stockholm, sweden: international joint conferences on artificial intelligence organization, jul. 2018, pp. 3926–3932. Doi: 10.24963/ijcai.2018/546.

[23] Ahmad rz, khan ms, bilal a, ali u, sattar rz. Effect of locus of control and depression among young adults in multan (pakistan). Journal of asian development studies. 2023 dec 30;12(4):684-92.

[24] A. A. Salsabil, e. B. Setiawan, and i. Kurniawan, "content-based filtering movie recommender system using semantic approach with recurrent neural network classification and sgd," *kinetik*, pp. 193–202, may 2024, doi: 10.22219/kinetik.v9i2.1940.

[25] J. Wang, k. Ding, l. Hong, h. Liu, and j. Caverlee, "next-item recommendation with sequential hypergraphs," in *proceedings of the 43rd international acm sigir conference on research and development in information retrieval*, virtual event china: acm, jul. 2020, pp. 1101–1110. Doi: 10.1145/3397271.3401133.

[26] A. Nurcahya and s. Supriyanto, "content-based recommender system architecture for similar e-commerce products," *jurnal informatika*, vol. 14, no. 3, p. 90, sep. 2020, doi: 10.26555/jifo.v14i3.a18511.

[27] M. R. Shahabi, department of communication engineering, university of sistan and baluchestan, m. Rezaei, department of communication engineering, university of sistan and baluchestan, f. Mohanna, and department of communication engineering, university of sistan and baluchestan, "content-based image retrieval for carpet e-commerce application," *ijictr*, vol. 13, no. 1, pp. 40–49, mar. 2021, doi: 10.52547/ijict.13.1.40.

[28] T. Wang, y. M. Brovman, and s. Madhvanath, "personalized embedding-based e-commerce recommendations at ebay," 2021, doi: 10.48550/arxiv.2102.06156.

[29] O. Dogan, "a recommendation system in e-commerce with profit-support fuzzy association rule mining (p-farm)," *jtaer*, vol. 18, no. 2, pp. 831–847, apr. 2023, doi: 10.3390/jtaer18020043.

[30] I. H. Arsytania, e. B. Setiawan, and i. Kurniawan, "movie recommender system with cascade hybrid filtering using convolutional neural network," vol. 10, no. 2, 2024.

[31] C. Yin, s. Ding, and j. Wang, "mobile marketing recommendation method based on user location feedback," *hum. Cent. Comput. Inf. Sci.*, vol. 9, no. 1, p. 14, dec. 2019, doi: 10.1186/s13673-019-0177-6.

[32] C. Ma, l. Ma, y. Zhang, j. Sun, x. Liu, and m. Coates, "memory augmented graph neural networks for sequential recommendation," 2019, doi: 10.48550/arxiv.1912.11730.

[33] K. Muhammad *et al.*, "fedfast: going beyond average for faster training of federated recommender systems," in *proceedings of the 26th acm sigkdd international conference on knowledge discovery & data mining*, virtual event ca usa: acm, aug. 2020, pp. 1234–1242. Doi: 10.1145/3394486.3403176.

[34] C. Udokwu, r. Zimmermann, f. Darbanian, t. Obinwanne, and p. Brandtner, "design and implementation of a product recommendation system with association and clustering algorithms," *procedia computer science*, vol. 219, pp. 512–520, 2023, doi: 10.1016/j.procs.2023.01.319.

[35] H. Simsek and m. Yeniad, "building a hybrid recommendation system for e-commerce," *j intell syst appl*, pp. 4–7, may 2022, doi: 10.54856/jiswa.202205190.

[36] P. Bellini, l. A. I. Palesi, p. Nesi, and g. Pantaleo, "multi clustering recommendation system for fashion retail," *multimed tools appl*, vol. 82, no. 7, pp. 9989–10016, mar. 2023, doi: 10.1007/s11042-021-11837-5.