IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Ingénierie des Systèmes d'Information
Vol. 29, No. 5, October, 2024, pp. 1999-2009
Journal homepage: http://iieta.org/journals/isi

A Progressive Approach for a Personalized Recommendation System Using Hybrid Deep Learning

Priyasha Gupta*, Arpit Deo, Safdar Sardar Khan, Ajeet Singh Rajput, Kriti Joshi, Ankita Chourasia

Department of Computer Science and Engineering, Medi-Caps University, Indore 453331, India

Corresponding Author Email: en20cs301325@medicaps.ac.in

https://doi.org/10.18280/isi.290531

Received: 26 December 2023
Revised: 16 August 2024
Accepted: 14 September 2024
Available online: 24 October 2024

Keywords:
e-commerce recommendation system, deep learning, re-ranking, hybrid recommendation system, user behaviour, cross domain analysis

ABSTRACT

A recommendation system is a type of statistical processing mechanism that anticipates and recommends goods, services, or content to users based on those users' preferences, interests, or activity history. The purpose of a recommendation system is to improve the customer experience and enable consumers in finding pertinent and worthwhile products or information. Our digital lives are now completely dependent on personalised recommendation algorithms, which assist users in finding information that suits their interests. In order to improve recommendation quality, this paper provides advanced deep learning techniques, personalised re-ranking model and hybrid recommendation approaches. In the initial stage, we retrieve context illustration using explicit, latent unstructured and latent structured data. Whereas a comprehensive study that categorises personalised recommendation models and investigates their advantages and disadvantages, measurement tools employed, and well-liked datasets. The following parameter are applied in our technique, analysing user behaviour, implementing fuzzy techniques on search data currently available and Applying context aware recommendation using deep learning techniques. Our end-user receives the best results produced after applying re-ranking model. The suggested model has 91% accuracy, 0.93 precision, 0.84 recall, and 0.87 F-Measure, according to our comparison of it with other models.

1. INTRODUCTION

E-commerce has grown along with online shopping and now contributes significantly to the global economy. E-commerce platforms use recommendation algorithms to enhance the user experience and assist customers in locating relevant products, with the inclusion of many more areas including food applications and shopping websites including everyday essentials to high-end fashion. Recommendation systems are a ubiquitous and successful paradigm in the retail industry. A title of article should be the fewest possible words E-commerce recommendation systems strive to provide consumers with personalized product recommendations based on their preferences, browsing patterns, previous purchases, demographic information, and other relevant data [1]. These initiatives improve user interaction, boost platform conversion rates, and help users locate the content they are likely to find interesting. These recommendation systems make use of artificial intelligence, algorithms, and data analysis to make the following recommendations. In a typical recommendation system, there are often two sorts of data used: rating or purchase behaviors' that represent the information that can be linked to the objects and users, and keywords or textural profiles that document the user-item interactions [2]. Overall, users and businesses benefit equally from e-commerce recommendation algorithms. Businesses may increase customer engagement, conversion rates, and revenue creation while users gain from personalized shopping experiences, effective product discovery, and increased satisfaction. E-commerce recommendation systems use concepts from Machine Learning and Deep learning.

Machine learning techniques are used by recommendation systems to analyse user data, spot trends, and produce tailored recommendations. Here are three machine learning methods that are frequently applied in recommendation systems Collaborative Filtering: To generate recommendations, collaborative filtering takes advantage of user preferences and behaviors [3]. It recognizes people with like tastes and suggests products that individuals with comparable profiles have enjoyed or purchased. This method does not explicitly require knowledge of item attributes and can be based on user-user similarity or item-item similarity. Content-Based Filtering [4] is a technique that recommends items based on the similarity of their content. [4]. This method of filtering places an emphasis on the properties of the items themselves. To construct item profiles, it examines item information like descriptions, categories, or tags. The algorithm suggests products that have qualities that the user has previously enjoyed based on the user's preferences. Hybrid Approaches: To increase suggestion accuracy and diversity, hybrid recommendation systems incorporate several strategies. In a hybrid method, for instance, collaborative filtering and content-based filtering may be combined. These systems take advantage of each technique's advantages to produce more precise and varied recommendations. The application of machine learning techniques in recommendation systems enables personalized and dynamic recommendations by continuously adapting to user feedback and learning from historical user data.

Deep learning algorithms are also used by recommendation systems to identify complex linkages and patterns in user data. Three prevalent deep learning approaches in recommendation systems are listed as follows: Deep neural networks can be used to learn non-linear representations of user-item interactions [1]. Examples of these networks include feed forward neural networks and multi-layer perceptron. These networks develop hierarchical representations using input features including user demographics, object qualities, and contextual data to produce precise suggestions. Recurrent neural networks (RNNs): RNNs operate effectively in recommendation systems with sequential data. They can represent time dependencies and the sequential character of user behavior [5]. RNNs can anticipate future user behaviour and offer individualized suggestions by considering the sequence of user interactions.

CNNs (convolutional neural networks), is frequently used in recommendation systems to extract information from item photos or textual data [6]. These networks may learn high-level representations of item features by using convolutional layers, enabling suggestions that are either visual or text-based. Deep learning techniques in recommendation systems enable the extraction of intricate patterns and latent features from user data, leading to more accurate and nuanced recommendations. In order to provide product recommendations, collaborative filtering activities require an autoencoder [7]. An autoencoder is a neural network that learns to encode inputs into a hidden (and typically low-dimensional) representation by copying input to output [8]. Given their ability to approximate any continuous function, neural networks are well suited for overcoming matrix factorization's drawbacks and boosting its expressiveness.

Traditional systems for recommendation have drawbacks, such as requiring a cold start for new users and items, The cold start problem occurs when new users or products are added into the system. Because matrix factorization relies on previous interactions between users and merchandise to reveal latent variables, it is difficult to create good predictions for fresh users who have not previously conversed with any items, or for novel products that have not yet had any interactions. Without enough data to learn from, the model cannot accurately place new users or things in the latent factor space, resulting in poor suggestions. Which are inefficient for leveraging massive data. The data amount is too vast to examine relationships, test hypotheses, and calculate values. As per current studies various machine learning and deep learning algorithms are used which includes Multi-layer perceptron based recommender system, Auto encoder based recommender system, CNN, GNN, RNN, IA CN, DHMR and many more to optimize the user experience and maximize user satisfaction [8]. This study aims to evaluate recent theoretical and practical notable accomplishments, emphasize limitations, and indicate future research areas in ML and DL for personalized recommender systems using hybrid systems.

The objectives of this work are: (1) To develop an artificial intelligence-based e-commerce recommendation system capable of producing high-quality, relevant recommendations for users. (2) Matrix factorization used to generate suggestions. (3) Cross domain recommendation which produces results from trending items. (4) Result from all the combined sets then undergoes transformer encoding. In section 3 we have explained the machine learning algorithms deeply with its experiments and analysis in section 4. We end with Conclusion and future work in section 5.

2. RELATED WORK

As an example of collaborative filtering, Dellal-Hedjazi and Alimazighi [1] first discussed the two recommendation methods in this study: content-based and demographic-based. Next, they prepared a presentation on neural networks, in particular deep learning, and the many designs of these networks. Based on the Movie lens dataset, which was split into two parts-one for learning the model and the other for validation-they created an MLP-style deep learning model. This recommendation engine may be implemented under a big data platform, tailored to various sorts of items (like books), and incorporated into an e-commerce site. According to Fanca et al. [9], the MovieLens20M dataset, which consists of 20 million user reviews, critic reviews, and movie ratings, was used to evaluate the system's performance. ML.NET, a framework of machine learning, an open-source material was used to develop a collaborative filtering-based recommendation system. The second model was developed using Microsoft's Azure Machine Learning Studio, which provided an altogether new way to develop machine learning solutions. Even while the initial pure collaborative filtering solution had a lower RMSE that does not necessarily mean it was superior. By carefully evaluating the type of data available and the problem that advice should address, the best recommendation method can be chosen. Unger et al. [10] describe three different types of contextual representations in their three deep context-aware models: explicit, latent, and hierarchically latent information. The suggested models, as demonstrated by experimental findings on three context-aware datasets. Modern context-aware approaches employed for various tasks, including rating prediction, creating top-k suggestions, and categorizing user input, were dramatically outperformed by ECAM, UCAM, and HCAM. Also, they presented that when the structured latent context is used in the deep framework of the recommendation system it outperforms other context-aware systems throughout all tasks. Abbas et al. [11] use experiments to validate the suggested approach for citation suggestion. There are a few restrictions, nevertheless, that demand more research. All rhetorical zones currently have the same weight when calculating similarity. This study has found that giving rhetoric zones dynamic weights will raise the rating finally of the suggested reading list. Additionally, the embeddings window is fixated at 300, but it can be changed to better assess its efficacy. In the current work, each rhetorical zone is given a single class label however, multi-class categorization might produce different results. Li et al. [12] discovered that HRM, their hybrid re-ranking model, outperformed the baseline of production. The graph embeddings in the content component perform best, particularly the author similarity depending upon soft matching. Users' most recent browsing may result in recommendations that are better than basic historical browsing however, similarities of popularity and impacts are insufficient to produce high-quality recommendations on their own. A limitation of their model is that they can’t confirm the effect of the model on users because of a lack of online evaluation techniques such as A/B testing. Another drawback is imposed by the production dataset where the articles of the candidates can only be re-ranked by the production system. As a result, if the inputs are of poor quality, their final suggestions will perform poorly. Jeong et al. [13] demonstrate that, compared to the current model, their suggested context-aware citation recommendation model significantly improves metrics such as Recall@K, MRR, and MAP. according to the result the model may achieve cutting-edge performance, recall@k improvement of more than 28%, and a mean average precision. The paper citation network, graph data is processed into a paper latent representation by the VGAE used in their framework's citation encoder. Performance gains over a BERT-based model are made possible by regularising the encoded paper network and the encoded context's combination. To solve this issue, they created and made available the FullTextPeerRead dataset. Gupta et al. [14] introduce a self-encoder-based multi-modal recommendation architecture in this study, which effectively combines textual, visual, and contextual data. This integration enhances support for suggestions and significantly improves video recommendation performance. The architecture consists of three essential parts, specifically: In order to extract context from textual input, a module that is a contextual feature based upon CNN is first used, Secondly, a fusion module multi-modal that combines visual features with contextual ones which combine the complementary information. in an efficient manner. And lastly, a recommendation module inspired self encoder that creates tailored suggestions using the fused features. Experimental findings on a number of real datasets with particular quirks serve as proof of the effectiveness of the proposed strategy. In this research paper, Zhang et al. [15] highlight several prototypes based on significant research and provide a classification approach for organizing and clustering existing papers. They also discussed the pros and cons of using deep learning analogies to build recommendation systems. Recommendation systems and their respective deep learning methods are a prevalent area of study nowadays. Every year a good number of novel methodologies and models are created in this field. This survey can give readers a thorough grasp of the most important elements of this subject, explain the most significant advancements, and provide some insight into the following research. Wang et al. [16] proposed an end-to-end, memory-network-based, context-sensitive citation recommendation model. The model uses bidirectional long short-term memory (Bi-LSTM) to learn representations of publications and citation contexts, respectively. In the distributed vector representations of citation contexts and journals, they jointly incorporate author information and citation relationship. To assess the effectiveness of their model, they also run tests on three real- world datasets. Additionally, it incorporates individualized author and citation relationships to produce high-quality, effective representations for papers and citation contexts. Results from experiments on the AAN, DBLP, and RefSeer datasets demonstrate the power of their model. Si et al. [17] analyse various word embedding techniques and look into how well they perform on four clinical concept extraction tasks. They contrast conventional word representation techniques with cutting-edge contextual representation techniques. Additionally, they contrast the effectiveness of pre-trained contextual embeddings utilizing a sizable clinical corpus with that of pre-trained models that are available off-the-shelf on open-domain data. Additionally, their findings demonstrate the advantages of embeddings by unsupervised pre-training on clinical text corpora, which outperform commercially available embedding models and produce new state-of-the-art performance across all tasks. Ma and Wang [18] create a novel paper recommendation system based on heterogeneous graph representation. Not only considers the diverse entities but also the user and paper proles are first extracted based on the paper's contents (i.e., title, keywords, abstract). Thirdly, two proximity metrics based on meta-paths are suggested. These measures assess the structural and neighbouring similarity of nodes in heterogeneous graphs. In the end, the paper recommendation is produced by comparing the user and paper feature vector similarities. For instance, the meta-paths employed in this study were created by hand. The ability to automatically build the meta-paths may be added in the future using some meta-pattern finding techniques. According to Deo et al. [19], the recommendation system has rapidly advanced, making it a vital tool for online ecommerce. The results of their comparison of the proposed model with the DeepCONN, IACN, DMHR, and GHRS models are unmistakable: their model outperforms them all for the supplied dataset and has a low error with 91% accuracy. The recommended model performed exceptionally well when precision recall, and F-measure were also examined. They contrast MAE values and AUC-ROC as well and conclude that the proposed model is more effective. The same could be further looked for in the proposed model. They want to overcome the issues with deep learning-based algorithms' typically high costs. Yu et al. and Lee et al. [20, 21] developed several techniques-including a conventional classifier, heuristic scoring, and machine learning-to construct a recommendation system and integrate content-based collaborative filtering using Co-Clustering with Augmented Matrices (CCAM). Insufficient data was obtained for this particular app because more than 50% of users clicked fewer than 10 times in 1.5 years. As a result, they encountered a difficulty known as the "cold-start problem" when analysing user data. They used web crawlers and frequent user analysis to improve our item-based data to gather enough purchase records, yielding F-scores ranging from 0.756 to 0.802. Crawlers are therefore used by us to gather data that helps us comprehend items better. The outcome demonstrates that user-based suggestions work better than item-based recommendations. Unger and Tuzhilin [22] believe that the limited explainability of the hierarchical model is one of the weaknesses of their work. It is challenging to label and annotate latent contextual situations since the construction of hierarchical latent contexts takes place in a latent space containing abstract numerical information. Furthermore, rather than using conventional ML metrics like RMSE, F-measure, etc., it is critical to assess the impact of supporting hierarchical latent contexts in recommender systems using business performance metrics like CTR, adoption and conversion rates, and increased sales and revenues. This is because it is still unclear whether advancements in rating prediction measures like RMSE translate into more effective recommendations. Ambalavanan and Devarakonda [23] conducted a thorough analysis using the Clinical Hedges dataset of abstracts and titles gathered from MEDLINE to determine how to employ the latest neural network models when the task of filtering articles involves various criteria. Overall, it was shown that while the single integrated model had the highest recall, the cascade ensemble had the highest precision and F-measure. In these designs, SciBERT served as a useful core model. The findings suggest that a cascade ensemble of criterion-specific models performs better in terms of precision and F-measure results when text categorization comprises numerous independent criteria. But the single integrated model regularly outperformed the others in terms of recall, indicating that it might be a preferred architecture for high-recall applications that can handle a little more noise.

3. PROPOSED WORK

The proposed system's purpose is to establish an e-commerce recommendation system based on artificial intelligence to produce high-quality relevant recommendations for users. Since distinct systems are implemented to evaluate exclusive suggestions; let us discuss how all these methodologies collectively produce the desired results. For the first method, user behavior such as users' interest is used, aided with matrix factorization to generate suggestions. It is a technique for dividing a big user-item interaction matrix into component matrices representing latent variables, or concealed trends in user preferences and item features. By learning these latent variables, the algorithm may anticipate a users' likelihood of interacting with an item based on comparable individuals or products. This combination enables the recommendation engine to offer more tailored suggestions that closely match the users' current interests. The second system emphasizes a cross-domain recommendation system, which produces results from trending items. It goes beyond the standard recommendation strategy by generating ideas from numerous unique domains or segments, rather than confining recommendations to a single domain. By recognizing popular items across multiple domains, the system may recommend what is presently popular, even if the user has no prior experience with that type of content.

Now the shown result is the log data which alludes to the recorded behavior of individuals on a platform, it includes precise information on their interactions over the course of time. This data comprises a variety of actions, including clicks, page visits, searches, buying decisions, and even time spent on certain content.

Which is then combined with a decision tree and matrix factorization to produce recommendations. Decision trees are used to categorize users relying on easily comprehensible criteria, and then matrix factorization is applied within each group to produce individualized suggestions. This method enables the system to detect both obvious correlations between parameters and more subtle, latent patterns in user behavior.

Applying context-aware framework to get popular items in social media. Following up, when the recently searched items are taken for profiling and matching items, the system generates a set of recommended items for the user. All the results when compiled together give a set with recommendations from all the different systems mentioned above. To build a highly refined recommendation set as a result for this combined set, it undergoes transformer encoding. Let us now discuss each system in detail.

3.1 User based recommendations with the help of deep neural networks

User Behaviour Analysis: In the suggested approach, the user's interactions and browser information, i.e., user's interest is used. The multilayer perceptron (MLP) is non-linearly activated feed-forward neural networks that consist of three or more layers which includes input layer, hidden layers, and an output layer, employing nonlinear activation functions in its neurons to record complex patterns in human behavior and item properties. By evaluating user and item data via these layers, MLPs learn to anticipate the chance of a user interacting with a specific item. This strategy is especially successful in situations where several variables influence user habits in multifaceted ways.

As a result, users' preferences for items are represented as a rating matrix to construct the relationship between users and items to locate users' relevant objects. The matrix factorization can be considered as a one directional model of latent vectors Eq. (1) *av* and *bi.*

Figure 1 shows the process starts with customer behavior analysis, which collects and analyzes data on users' interactions with products such as clicks, views, ratings etc. This analysis aids to comprehend user interests and patterns in user-item relationships, which are the foundation for creating correct suggestions.

This behavioural data is then sent into a deep neural network (DNN), which has levels of processing to capture complicated, non-linear interactions between users and things.

The multilayer perceptron (MLP) continues to process the data by passing it through several layers of neurons with non-linear activation functions. The MLP helps to develop deep, latent representations of persons and items, as well as to detect hidden elements that influence user preferences.

Following the MLP, matrix factorization is used to break down the user-item interface matrix into lower-dimensional matrices. This stage aids in identifying latent elements that are not readily apparent, such as a users' preference for specific genres or product attributes.

Finally, the system analyses this data to provide recommendations based on each users' specific tastes. The flow chart describes a complete method of recommendation that incorporates user behavior analysis, deep learning, and matrix factorization, leveraging both explicit and implicit signals to improve the relevance and efficacy of suggestions presented to users.

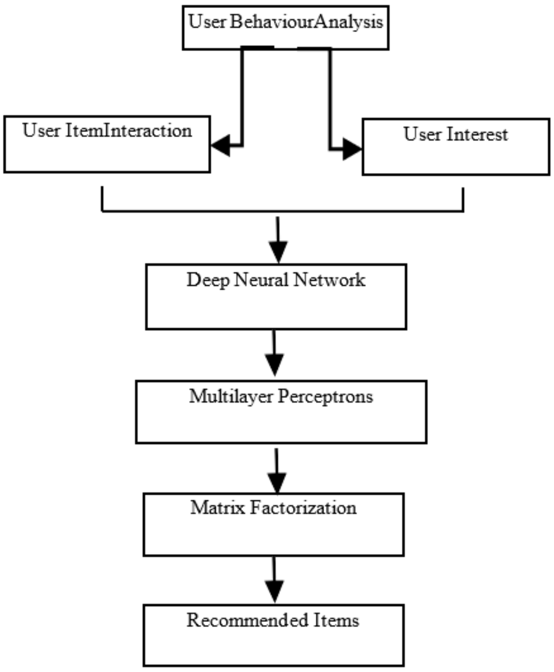

Figure 1. Recommendations based on user behavior

$$x_{vi} = g(v, i || a_v, b_i) = a_v^t b_i = \sum k \qquad (1)$$

where,

x_{vi}: $Real\ valued\ latent\ vector\ for\ user$

bi: $Real\ valued\ latent\ vector\ for\ item$

k: $Latent\ space$

The bi-directional user and item latent factors' communication considers that each direction of the latent space is linearly added with the same load which results in a filtered list of items produced from user behavior analysis for certain items in the e- commerce system. This model is further formulated as:

$$xvi = \sigma\left(f^t(\phi out)\right) \qquad (2)$$

In order to describe user-item latent structures, Eq. (2) suggests that the model combines linearity from matrix factorization and non-linearity from neural networks.

3.2 Recommendations based on log data using active learning techniques

Every user-item pairing in a recommender system is essential for determining a user's preferences and has a big influence on how well the system works. The problem of data scarcity demonstrates that the more user ratings a system collects, the better it will perform in making recommendations.

Audit Trail: The web server keeps a log file called the audit trail. This online access server log describes how, and which pages are accessed by various web users. It is an orderly record that contains extensive details on the order in which operations or transactions occur within a system. This record contains information such as the algorithms utilized, the parameters implemented, the user behaviors assessed, and the recommendations issued. It is vital for transparency, accountability, and compliance.

Recommender systems are now more accurate and efficient because of the implementation of active learning, which helps them choose the most representative things to give to consumers. Rapid profiling of users and ratings is often achieved through the rating impact analysis and bootstrapping process.

Figure 2 shows the method commences with user audit trails, which provide a thorough record of previous interactions, judgments, and suggestions, providing insight into how the system operates.

This data is paired with user ratings, which represent user feedback on things, to determine the following stages.

The exemplary item stage entails choosing crucial items that embody the user's preferences or are necessary in interpreting their behavior. These things are utilized in active learning, a procedure in which the system intelligently chooses the information points to prioritize in order to increase its forecast accuracy.

Following that, rating effect analysis/bootstrapping is used to determine how modifications in user ratings or the addition of new data affect the system's suggestions. Bootstrapping techniques can be used to evaluate the dependability of these adjustments, guaranteeing that the recommendations are strong and not unduly sensitive to minor differences in data.

This results in an improved comprehension of user behavior, enabling the system to tailor its recommendations based on more precise and complete information. Finally, the procedure generates revised recommendations based on the audit trail, to guarantee the recommendations are not only individualized but also consistent with the system's overall aims of openness and accountability.

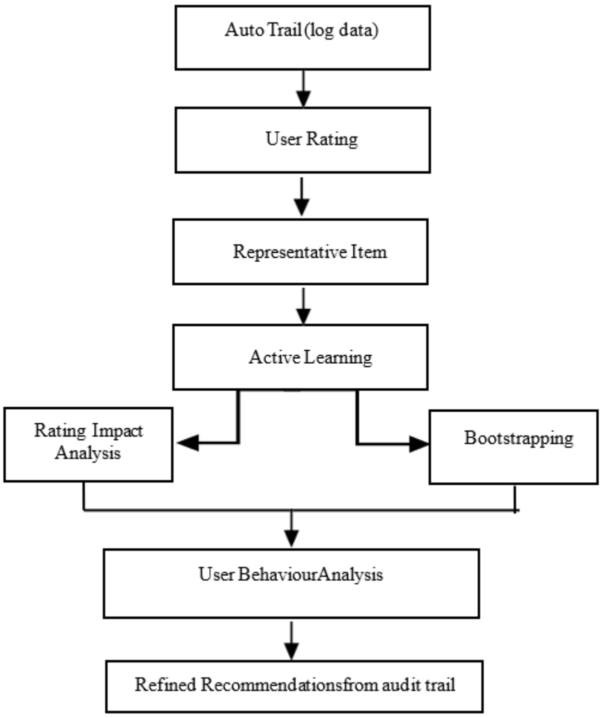

Figure 2. Recommendations using active learning with log data as a dataset

We employ decision trees for the model shown in Figure 3 because they produce a list of suggested items at their lead nodes. The recommendation system may scale by reducing the amount of searching necessary while creating a user's personalized suggestion list utilizing the tree. Similar to a tree probability formulation, a max heap is employed. For each user u, each non-leaf node x in level j fulfils Eq. (3) as follows:

$$x_0\{x'schildrennodeninlevelj + 1\}^{r(j+1)}(u)/a(j) \qquad (3)$$

where,

$r^{(j)}$(x|u)-probability that user u prefers

$a^{(j)}$-normalization for j

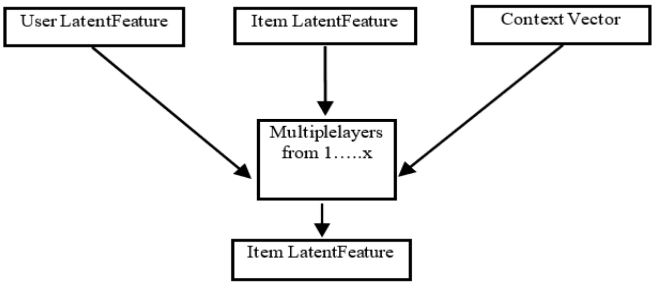

Figure 3. Context aware recommendation using deep learning

3.3 Recommendation based on social media popularity based on context aware framework using deep learning

A context-aware recommendation system establishes linkages among context, things, and users. Here, we start by creating a numerical vector of all prominent instances, to which we then add compressed latent contextual embeddings. Finally, we make systematic and implicit use of hierarchical contextual information. Figure 4 represents the framework.

We extracted context representation from apparent, implicit unstructured, and implicit structured data. To create a context-dependent vector with the form context = [x1, x2,..., xT], We initially normalize the context-relevant feature values to a 0–1 scale and turn conventional qualities into binary traits.

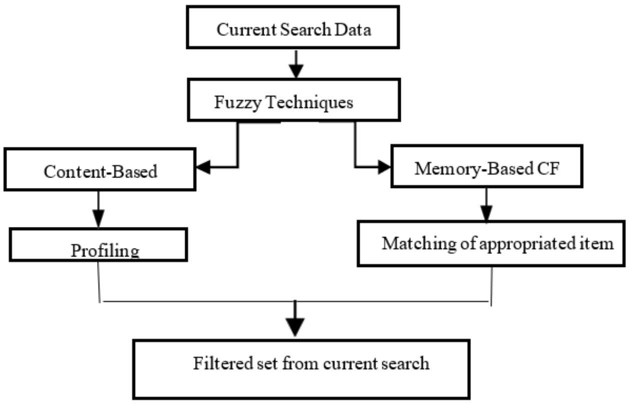

Figure 4. Recommendations from the current search dataset

In addition to unstructured latent contextual data, which is a compact representation including latent L values (LT) derived from hidden layers, explicit contextual representation extraction involves employing all of the available (raw) T contextual characteristics. This latent contextual information is utilized for categorization, top-recommendation, along with rating prediction. A compact feature representation is produced by the auto-encoder approach by passing the initial input values through a low-dimensional layer once again. In order to understand the structure of latent variables and the significance of relationships between them, we deploy a structured latent contextual representation.

For the recommendation process, we use a structured latent contextual model that consists of a set of contextual factors (cluster Identification) at various granularity levels that are generated via a hierarchical tree. A hierarchical model is created by automatically classifying an extensive set of unstructured implicit contextual vectors into a limited number of clusters, each representing an implicit contextual event.

To acquire structured latent contextual vectors, we primarily extract compressed latent contextual vectors from an AE's bottleneck layer, and then use the unstructured latent contextual vectors to construct a hierarchical tree at the resolution level. In order to combine contextual vectors with similar latent qualities that are pertinent to a certain context or organisational latent, we use automated agglomerative hierarchical clustering (AHC) to apply the k- means algorithm approach and assess how many ambient scenarios (clusters) there may be. It is a bottom-up clustering technique that can be applied in recommendation systems for combining comparable individuals or objects based on their features or interactions. In AHC, each client or item is assigned to its own cluster, which is recursively integrated based on resemblance until a certain number of clusters or a stopping threshold is met.

The path taken by raw information, or latent context-dependent vector, from a tree's top to a leaf. Then, we compile user reviews or ratings for each item and standardize their representation as one-hot vectors by index ID. The model is then trained using the data we have gathered about items, users, and contexts. We make advantage of ECAM, which gathers all latent unstructured contexts and feeds them into a neural CF model. In order to add unstructured latent contexts, we secondly employ the UCAM model. The HCAM model, which encompasses the hierarchical latent contexts observed in neural-CF models, is the last one we use. We amalgamate the contextual vector c to the client u and product i integration to develop a new nonlinear function spanning all three elements-objects, people, and contexts. In order to automatically assess how context impacts the predicted target as a whole, the neural framework considers context. In order to forecast which class would have the highest likelihood in a classification task, we first calculate the likelihood of each class label (categorical value) using the SoftMax activation function.

$$U{:}\,Set\ of\ Users(U = u1, u2, \dots, um)$$
$$I{:}\,Set\ of\ Items(I = i1, i2, \dots, in)$$

R: Matrix of user item ratings

C: Matrix of contextual information, where each data point cui represents the contextual information

We can modify the factorization equation as in Eq. (4):

$$R \approx U^{*}I^{T} + C^{*}H \tag{4}$$

The latent elements connected to the contextual information are captured by the matrix H. Through an optimisation procedure that reduces the difference between the estimated ratings derived from the factorization and the real ratings, H may be discovered.

3.4 Recommendations based on current search data using fuzzy techniques

Fuzzy collection and fuzzy link concepts can be helpful for addressing information uncertainty and can be used to recommender systems [24]. As shown in Figure 4, the two main categories of fuzzy recommendation approaches are memory-based CF recommender systems with fuzzy methods and content-based recommender systems with fuzzy techniques. This system can be enhanced through the application of fuzzy methods, which improve the handling of uncertainty and imprecision in user preferences. The dataset being utilized is the most current data that was searched.

These strategies can increase accuracy in specific areas by matching consumer desires with the services offered and reducing the inevitable noise of uncertainty and producing refined recommendation sets.

In Figure 4, it starts with the existing search data that gathers the user's present interests depending on their most recent queries or interactions with the system.

Next, fuzzy approaches are used to deal with the inherent unpredictability and complexity in client search activity. Fuzzy logic enables the system to make more versatile and subtle decisions by taking into account degrees of significance rather than strictly labelling items as relevant or irrelevant.

Following that, the system employs either content-based or memory-based collaborative filtering (CF). Content-based filtering suggests things like the ones with which the user previously communicated with respect to item qualities. Memory-based collaborative filtering, on the other hand, recommends products based on the actions of users with similar tastes, utilizing the expertise of the community to forecast outcomes.

The recommendations are further refined by profiling/matching applicable items, in which the algorithm fits possible ideas with the client's already defined profile or inclinations.

Finally, the procedure ends with a filtered set from the currently selected search, which displays a carefully selected group of objects that pertain to the clients' current search context and general preferences.

Rt maps from the rewards/scores ri. We now use the previous calculation to determine the end result and focal point of the entire list of recommendations:

$$pt = x \sum \gamma^{k-1} u^{k} \tag{5}$$

k=1

k-Order of each item

K-total items

$$\gamma \in (0,1]$$

Let us find the Pearson correlation coefficient, which computes the similarity between two users, v_x and v_y, based only on their common ratings.

$$corr(v_x, v_y) = \frac{\sum sgx\,,y(p_{(x,y)} - m_x)(p_{(x,y)} - m_y)}{\sqrt{\sum(p_{(x,k)} - m)^2 (p_{(y,k)} - m)^2}} \tag{6}$$

where,

px - Ratings of users vx for kth item

pbi -Ratings of users vy for kth item m-average ratings of user v

my-Average rating of user vy

In Figure 4, Current search data-based recommendation algorithm using fuzzy techniques

The cosine of angle will be:

$$cos(v_x, v_y) = \frac{\sum_p x\,, kp_y, k}{\sqrt{\sum_{p_{x,k}^2} p_k^2}} \tag{7}$$

The model is forced to arrange things that the user may order in the top of the suggested list because the idea from the aforementioned Eqs. (5)-(7) is that the final result and focus point on the top of the recommended list has a bigger contribution to the overall outcomes.

These sets are further augmented to form a quality recommendation set for the users.

3.5 Recommendations based on a collective set of generated recommendations using a personalized Re-Ranking model

A re-ranking model in a recommendation system is a sophisticated strategy for refining the sequence of the first set of recommendations, to make sure the most relevant things are given preference for the user. After the system generates an initial group of recommendations using typical approaches such as collaborative filtering or content-based filtering, the re-ranking model reorganizes the list depending on further factors that were not completely assessed in the original ranking [25].

The input layer receives a combined list of things made up of the outcomes of the earlier approaches covered in this work, which it uses to create comprehensive representations of all the items in the original list and send them to the encoding layer. Raw feature matrix X serves as a representation for the first sequential list of fixed lengths. For each item i in S, a row in X stands in for the raw feature vector xi.

Input Layer: The raw matrix and the customized matrix PV are combined to create an intermediate matrix E' for a user-specific encoding function. Additionally, add a position embedding PE to E' to take advantage of the sequential information in the original list.

Encoding Layer: In order to fulfil the combination of mutual influences, a transformer encoder is used, because, it directly represents the reciprocal influences of any two items, regardless of their distances. Every element in this encoding module's N-block structure has an attention module and a Feed-Forward Network (FNN). Eq. (8) describes how the attention layer works.

$$Attention(A, B, C) = softmax\left(\frac{AB^T}{\sqrt{d}}\right)C \tag{8}$$

where,

A-queries matrix

B-keys matrix

C-values matrix

D-dimensionality of matrix K table

Proposed algorithm

Output Layer: This layer aims to produce a score for each item I = i1, ……, in. Here, a linear layer and a SoftMax layer are utilized to determine how an item-pair's impacts interact. Eq. (9) contains the score formulation.

$$\begin{aligned} Attention(i) &= P(X, PV; \theta) \\ &= softmax\left(A^{(o_x)} Q^F b^F\right), i \in S \end{aligned} \tag{9}$$

where,

A-the output of N_x blocks

Q-projection matrix

B^F-bias term

o-total items

This will generate a probability of clicking for each item to generate a score. The set items with the relevant scores will be selected and recommended to the user. Based on the analysis performed so far, we can claim that the proposed personalized hybrid e-commerce recommendation system outperformed existing approaches.

The proposed e-commerce recommendation system overcomes the drawbacks and performs multiple filtering to produce rectified recommendations.

4. EXPERIMENT AND ANALYSIS

4.1 Dataset

So far it is known that the system uses statistical algorithms and analysis for various datasets. We have used web spider bot data from *Amazon* from McAuley et al. [26], whose attributes are shown in Table 1, Originally, this bot data consists of twenty distinct product categories; however, for our experiments, we choose six out of these that are based on user ratings and reviews. The dataset specifies four sorts of linkages between items A and B given a pair of them. Based on the following, item B is considered a replacement for item A: (1) users who looked at item A also looked at item B, or (2) users who looked at item A eventually bought item B. Furthermore, an item B is supplementary to item A, if: (1) users who purchased A also purchased B, or (2) users commonly purchased A and B together, in order to produce the dataset for our tests, we carry out some basic pre-processing operations, such as stemming, discarding items less than 20 reviews, etc.

Table 1. Amazon dataset [19]

Category	User	Items	Reviews	Edges
Weblog files (Electronics)	4.25M	498K	11.4M	7.87M
User Feedback (Women's Clothing)	1.82M	838K	14.5M	17.5M
Social media content (Music)	1.13M	557K	6.40M	7.98M
All	7.20M	1.89M	31.90M	33.35M

4.2 Baseline

We compare the proposed model with numerous representative and cutting-edge baseline procedures for link prediction of an item:

Deep Learning (DL): This method is used to treat cold start.Deep learning uses recommendation systems by taking advantage of their capability to model complicated relationships and develop rich interpretations from numerous sources of data. Numerous formulas are used to generate ratings of each item [1]. However, our proposed model outperforms DL in many aspects which is illustrated in the user study section.

Graph Neural Networks (GNN): A unified heterogeneous graph is used to connect two sources of information. With the help of this information, a graph neural network is designed by simultaneously augmenting information on these heterogeneous graphs, to encode into user and item representations [24]. However, our suggested model surpasses GNN in many ways, as demonstrated in the user study section.

IA CN model: This model feeds item and user records into an incorporating layer, which is globally interlinked layers. A CNN layer then gets it after that. For efficient manufacture, they additionally feature an extra Attention Layer. The final result will be suggested after factorization is complete [26, 27]. However, our suggested model outshines IA CN model in a number of ways, as demonstrated in the user study section.

DHMR Model: In this paper, the DMHR model-a hybrid recommendation technique built on the synthesis of viewpoints from many sources and in-depth emotional analysis-is developed. On user behavior preferences a hybrid recommendation system combines the content-based recommendation approach and the post-based collaborative filtering recommendation strategy, with a focus on deep semantic analysis and emotional mining of text data. The post's natural language narrative has also been extracted [28].

4.3 Testing

In this section, we evaluate our model's accuracy, precision, recall and F-measure to the baseline models.

(1) Precision: It is defined as the ratio of recommended items that are pertinent to the user among all items advised. In another way, precision measures the quality of suggestions by demonstrating the amount of the suggested items are genuinely useful or interesting to the user. The proportion of True Positives to the total of True Positives and False Positives is what defines this. We compare the proposed model with the following models: DL, GNN, IACN, and DHMR according to Table 2, Figure 5. However, our suggested model surpasses other models as demonstrated.

(2) Recall: Recall is an important evaluation parameter in recommendation systems since it evaluates the system's capacity to determine all relevant items for a user. Recall is described as the proportion of relevant items that were adequately recommended out of the entire quantity of available items. By dividing the total number of Positive samples by the percentage of Positive samples that were correctly classified as Positive, the recall is calculated. Table 3, Figure 6, shows recall values of each model. In which our proposed model outperforms from other baseline models.

$$RECALL = \frac{TRUEPOSITIVE}{TRUEPOSITIVE + FALSEPOSITIVE}$$

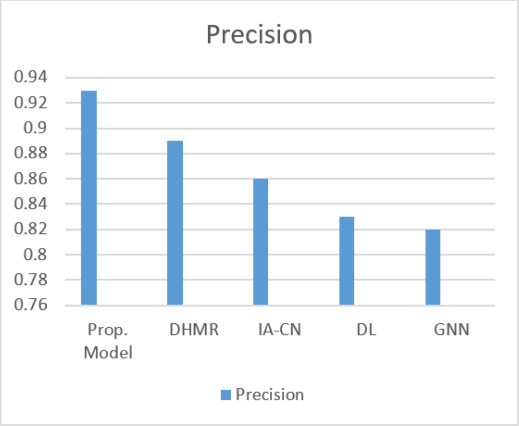

Figure 5. Precision

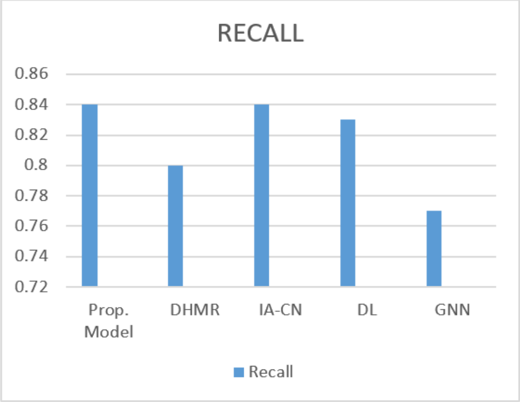

Figure 6. Recall

Table 2. Precision

Model	Precision
Proposed Model	0.93
DHMR	0.89
IA-CN	0.86
DL	0.83
GNN	0.82

Table 3. Recall

Model	Recall
Proposed Model	0.84
DHMR	0.80
IA-CN	0.84
DL	0.83
GNN	0.77

Table 4. F-Measure

Model	F-Measure
Proposed Model	0.87
DHMR	0.84
IA-CN	0.84
DL	0.81
GNN	0.79

Table 5. Accuracy

Model	Accuracy
Proposed Model	91%
DHMR	81%
IA-CN	75%
DL	71%
GNN	71%

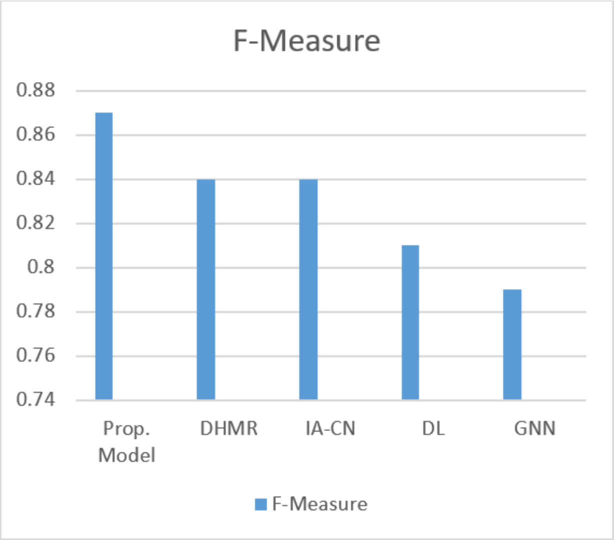

Figure 7. F-Measure

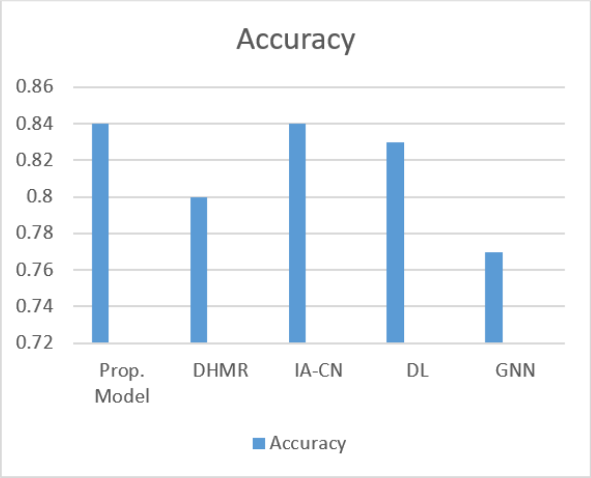

Figure 8. Accuracy

(3) F-Measure: F-measure, commonly known as F1 score, is a metric utilized within recommendation systems to assess the equilibrium of precision and recall. It is the harmonic mean of precision and recall, yielding a single score that takes into account both the correctness of recommended items (precision) and the system's capacity to retain all relevant information (recall). It covers both characteristics. Table 4, Figure 7 The results demonstrate that the suggested model outperforms existing models by 87.05%.

$$F - MEASURE = \frac{PRECISION * RECALL * 2}{(PRECISION + RECALL)}$$

(4) Accuracy: Accuracy is a typical evaluation statistic in recommendation systems that assesses the overall preciseness of the system's recommendations. It is described as the percentage of accurate recommendations among the total number of things considered. In essence, accuracy measures how frequently the system's recommendations correspond to the real user choices. The proposed model gives an accuracy of 91% as shown in Table 5, Figure 8. Which is greater from the accuracy of other compared models.

5. CONCLUSION AND FUTURE WORK

A recommendation system in e-commerce is an innovation that analyses user data and product details to propose appropriate products or goods to users. A notable development in the field of recommendation systems is the creation and application of a progressive approach for a personalized recommendation system utilising hybrid deep learning technique. This method gives customers recommendations that are more precise and tailored to their needs by combining the benefits of progressive tactics and deep learning approaches. It takes advantage and benefits of deep learning, fuzzy techniques, deep neural network an re- re-ranking model by combining them into a hybrid model that improves user experience and recommendation quality. The findings of our comparison of the proposed model with the DHMR, IA-CN, DL and GNN models are clear: Our algorithm excels all of those for the offered dataset, with a minimal error of 91.05% efficiency. We also compared F- measure, recall, and precision; in all cases, the recommended model fared exceptionally well and we found that the suggested model is more effective. We should make an effort to incorporate diversity as a goal into our re-ranking model.

In the future this avenue will be investigated more. One feasible approach for future research is to create re-ranking algorithms that unambiguously account for variability by incorporating metrics like intra-list diversity, which gauges the variance of recommended items. This could entail developing hybrid models that combine traditional relevance values with diversity values throughout the re-ranking phase.

Another area of future research could be user-centric variance models that adjust for specific customer preferences for diversity. The various users have various preferences for the level of diversity in their recommendations-some may value a broader range of possibilities, while others may prefer more specific suggestions.

REFERENCES

[1] Dellal-Hedjazi, B., Alimazighi, Z. (2020). Deep learning

for recommendation systems. 2020 6th IEEE Congress on Information Science and Technology (CiSt), Agadir-Essaouira, Morocco, pp. 90-97. https://doi.org/10.1109/CiSt49399.2021.9357241

[2] Mu, Y., Wu, Y. (2023). Multimodal movie recommendation system using deep learning. Mathematics, 11(4): 895. https://doi.org/10.3390/math11040895

[3] Paravttikar, S., Parasar, D. (2020). Recommendation system using machine learning. In Proceedings of the International Conference on Recent Advances in Computational Techniques (IC-RACT) 2020. http://dx.doi.org/10.2139/ssrn.3702439

[4] Roy, D., Dutta, M. (2022). A systematic review and research perspective on recommender systems. Journal of Big Data, 9(1): 59. https://doi.org/10.1186/s40537-022-00592-5

[5] Rahman, M.M., Prity, S.K., Bari, Z.A. (2021). Machine Learning approach for Item-basedMovie Recommendation using the most relevant similarity techniques. In 2021 3rd International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA), Ankara, Turkey, pp. 1-4. https://doi.org/10.1109/HORA52670.2021.9461381

[6] Narayan, S. (2020). Multilayer perceptron with autoencoder enabled deep learning model for recommender systems. Future Computing and Informatics Journal, 5(2): 96-116. https://doi.org/10.54623/fue.fcij.5.2.3

[7] Ferreira, D., Silva, S., Abelha, A., Machado, J. (2020). Recommendation system using autoencoders. Applied Sciences, (Switzerland), 10(16): 5510. https://doi.org/10.3390/app10165510

[8] Zhang, Q., Lu, J., Jin, Y. (2021). Artificial intelligence in recommender systems. Complex & Intelligent Systems, 7(1): 439-457. https://doi.org/10.1007/s40747-020-00212-w

[9] Fanca, A., Puscasiu, A., Gota, D.I., Valean, H. (2020). Recommendation systems with machine learning. In 2020 21th International Carpathian Control Conference (ICCC), High Tatras, Institute of Electrical and Electronics Engineers Inc., Slovakia, pp. 1-6. https://doi.org/10.1109/ICCC49264.2020.9257290

[10] Unger, M., Tuzhilin, A., Livne, A. (2020). Context-aware recommendations based on deep learning frameworks. ACM Transactions on Management Information Systems (TMIS), 11(2): 1-15. https://doi.org/10.1145/3386243

[11] Abbas, M.A., Ajayi, S., Bilal, M., Oyegoke, A., Pasha, M., Ali, H.T. (2024). A deep learning approach for context-aware citation recommendation using rhetorical zone classification and similarity to overcome cold-start problem. Journal of Ambient Intelligence and Humanized Computing, 15(1): 419-433. https://doi.org/10.1007/s12652-022-03899-6

[12] Li, X., Chen, Y., Pettit, B., Rijke, M.D. (2019). Personalised reranking of paper recommendations using paper content and user behavior. ACM Transactions on Information Systems (TOIS), 37(3): 1-23. https://doi.org/10.1145/3312528

[13] Jeong, C., Jang, S., Park, E., Choi, S. (2020). A context-aware citation recommendation model with BERT and graph convolutional networks. Scientometrics, 124: 1907-1922. https://doi.org/10.1007/s11192-020-03561-y

[14] Gupta, R., Patel, A., Wilson, K., Brown, K. (2023). Context-aware multi-modal recommendation model based on deep auto-encoder. Research Article. https://doi.org/10.21203/rs.3.rs-2801409/v1

[15] Zhang, S., Yao, L., Sun, A., Tay, Y. (2019). Deep learning based recommender system: A survey and new perspectives. ACM Computing Surveys (CSUR), Association for Computing Machinery, 52(1): 1-38. https://doi.org/10.1145/3285029

[16] Wang, J., Zhu, L., Dai, T., Wang, Y. (2020). Deep memory network with BI-LSTM for personalized context-aware citation recommendation. Neurocomputing, 410: 103-113. https://doi.org/10.1016/j.neucom.2020.05.047

[17] Si, Y., Wang, J., Xu, H., Roberts, K. (2019). Enhancing clinical concept extraction with contextual embeddings. Journal of the American Medical Informatics Association, 26(11): 1297-1304. https://doi.org/10.1093/jamia/ocz096

[18] Ma, X., Wang, R. (2019). Personalized scientific paper recommendation based on heterogeneous graph representation. IEEE Access, 7: 79887-79894. https://doi.org/10.1109/ACCESS.2019.2923293

[19] Deo, A., Jaisinghani, R., Gupta, S., Khan, S.S., Soni, A., Gehlot, K. (2023). Stratified advance personalized recommendation system based on deep learning. Ingénierie des Systèmes d'Information, 28(1): 189-196. https://doi.org/10.18280/isi.280121

[20] Yu, M., Quan, T., Peng, Q., Yu, X., Liu, L. (2022). A model-based collaborate filtering algorithm based on stacked AutoEncoder. Neural Computing and Applications, 34: 2503-2511. https://doi.org/10.1007/s00521-021-05933-8

[21] Lee, C.Y., Lin, C.L., Chang, H.T. (2024). Technologies and applications of artificial intelligence. 28th International Conference, TAAI 2023, Yunlin, Taiwan, 2074. https://doi.org/10.1007/978-981-97-1711-8

[22] Unger, M., Tuzhilin, A. (2019). Hierarchical latent context representation for CARS. IEEE Transactions on Knowledge and Data Engineering, 32(1): MAY 2019.

[23] Ambalavanan, A.K., Devarakonda, M.V. (2020). Using the contextual language model BERT for multi-criteria classification of scientific articles. Journal of Biomedical Informatics, 112: 103578. https://doi.org/10.1016/j.jbi.2020.103578

[24] Pei, C., Zhang, Y., Zhang, Y., Sun, F., Lin, X., Sun, H., Wu, J., Jiang, P., Ge, J., Ou, W., Pei, D. (2019). Personalized re-ranking for recommendation. arXiv:1904.06813. https://doi.org/10.48550/arXiv.1904.06813

[25] Liu, W., Liu, Q., Tang, R., Chen, J., He, X., Heng, P.A. (2020). Personalized Re-ranking with item relationships for E-commerce. In Proceedings of the 29th ACM International Conference on Information & Knowledge Management, pp. 925-934. https://doi.org/10.1145/3340531.3412332

[26] McAuley, J., Pandey, R., Leskovec, J. (2015). Inferring networks of substitutable and complementary products. In Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Association for Computing Machinery, New York, NY, USA, pp. 785-794. https://doi.org/10.1145/2783258.2783381

[27] Ming, F., Tan, L., Cheng, X. (2021). Hybrid

recommendation scheme based on deep learning. Mathematical Problems in Engineering, 2021(1): 6120068. https://doi.org/10.1155/2021/6120068

[28] Jiang, L., Liu, L., Yao, J., Shi, L. (2020). A hybrid recommendation model in social media based on deep emotion analysis and multi-source view fusion. Journal of Cloud Computing, 9: 1-16. https://doi.org/10.1186/s13677-020-00199-2