Hybrid advice gadget using context aware-neural collaborative filtering

Raja Mani.V[1], B. Tandle [2] Venkat Ramanan. G[3]
[1]Assistant Professor, Department of Computer Science and Engineering, RGIT Mumbai MH,
[2] Research Scholar, Department of Computer Science and Engineering, , MERIT Mumbai MH,
[3] Assistant Professor, Department of Computer Science and Engineering, RGIT Mumbai MH,

Abstract: In recent years, online based learning has created a global impact in the field of education. E-Learning uses internet and computer to easily access vast learning resources. In a wide range, it efficiently delivers consistent content to all the target audience. Yet not all learners who take advantage of this have same level of interest and ability to capture the knowledge. Though it is panacea to explore different topics in the education field, this has also potential pitfalls when the apprentices are not aware of the path to choose in their respective field. Hence the recommended system comes to the limelight. The Recommender systems recommend diverse content to different learners depending on their interests or preferences. In this paper, we propose a hybrid model incorporating Context aware filtering and Neural Collaborative Filtering called Context Aware-Neural Collaborative Filtering (CA-NCF) to recommend desirable resources to the target audience. This proposed method considers context information of learners as beginner, intermediate and master. The result of CA-NCF is compared with User based collaborative filtering (UBCF) and Item based collaborative filtering (IBCF). The performance measure like recall, ROC curve and precision of the proposed model illustrates quality and decision-making process.

Keywords: Recommender systems, E-Learning, Context Aware Filtering, Neural Collaborative Filtering, Education

1. Introduction

E-Learning covers huge amount of educational resources. Due to material overload, they incur more time for selecting the essential materials than educating oneself. To the learner's choice recommender system outcomes this crisis through the process of filtering and recommending the related resources to the learners. This evokes appropriate products in e-commerce sites, news, books, courses and videos/movie recommendation in Over-The-Top (OTT). Diverse learners have diverse demands of materials and interest. Recommender system (RS) cordially functioning based on the historical data and commodities relevancy of items i.e., the number of times viewed by the users. Personalized E-Learning RS offer suggestions for relevant and essential learning materials to the learners [28][29]. Almost all the RS operates on the conception of similarity between materials or users. There exist three vital similarity measurements such as Euclidian distance, Cosine distance and Pearson distance. Out of those Pearson's correlation coefficient is a widely known correlation coefficient which is considered between two variables as the covariance of the 2 variables divided by the multiplication of their respective standard deviations. The classification of the RS, Content Based Filtering (CBF), and Collaborative Filtering (CF) as in figure 1 are discussed in the following section.

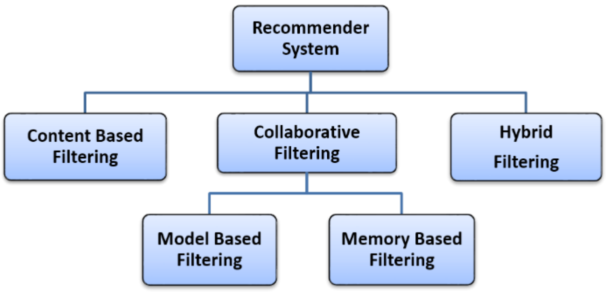


Figure 1. Types of Recommender system

1.1 Content Based Filtering

Content Based Filtering as in figure 2 is structured on the preferences of users and characteristics of materials. For instance, while a user is having a notion for a book, similar books are recommended to the user [30]. CBF focuses on user profiles which are advanced based on historical user's data to put forward suggestions for the user. Inverse Document Frequency (IDF) and Term Frequency (TF) are used for the retrieval of data in the content-based filtering. TF is number of a word in document while IDF is opposite of the document frequency in whole document. It is more like the classical machine learning that works based on implicit rating or explicit click on link.

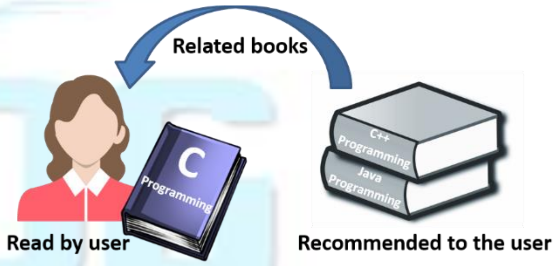


Figure 2. Context based Filtering

1.2 Collaborative Filtering

Collaborative Filtering (CF) as in figure 3 is serving based on the history of user preferences for items in the format of user/ item rating matrix. For example, while user1 and user2 both are having same interest, considering user1 reads a book which is also recommended to the user2. The module of carrying the task is based on both explicit rating that are to be mentioned as purchase history clicks and views such as score for items in the context of rating from 1 to 5. CF can be classified into two types such as model based approach and memory-based approach. Model based approach utilizes the machine learning technique to find the ratings for the unrated items by means of Neural Nets, Matrix Factorization, Singular Value Decomposition (SVD) and PCA. Memory based model find the similar users and calculate the weighted average rating for unrated items. The example for memory-based model is K-Nearest Neighbor (KNN). It uses the cosine similarity and Pearson correlation for finding the similarity between the users. In the case of CF technique Cold start crisis is the bottleneck. i.e., if a quite novice user or item peeps into the system, the recommendation becomes incorrect since of unavailability of user/material matrix. Two modules of Collaborative Filtering are also put forth. User-based CF measure the similitude amid the users, Item-based CF gauges the resemblance amid the items and rating of target users whereas Hybrid Filtering is the mixture of Content Based Filtering and Collaborative Filtering. The two key areas of CF are latent factor and neighborhood methods.

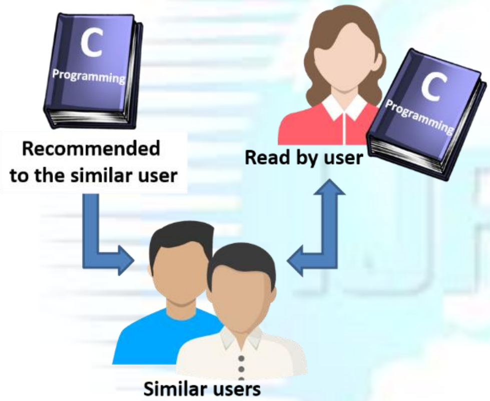


Figure 3. Collaborative Filtering

2. Literature survey

In [10] CF technique is used for personalizing the items using RF. Here CF works by assessing the rapport amid the users and inter-dependencies among items. It can be either based on items or user to build a database of connections between them. The algorithms used for this filtering are Cosine based Correlation and Pearson Correlation. Also, in [9] to recommend learning courses that are suitable to the students as per the learning level collaborative systems was put forth. Keeping in sight of the difficulties faced by students, CF helps to improve the learning performance by recommending the courses that are more fit to respective students' level of learning. To make the correlation between student skills and their profiles, the curriculum development, student skill model, Delphi analysis were exploited. The result analysis of CF methods indicates that the students have better result with satisfaction rather than facing challenging time with time trial and error methods. RS by simple Bayesian model [13] combining user-based and item-based collaborative filtering to improve the performance of predictions. The similarity between users or items is calculated from negative and positive ratings separately. Among these methods Item-User combination produced better performance on movie dataset. RS for E-Commerce site by SVD [2] for dimensionality reduction generated better result than traditional collaborative filtering methods. Here 2 experiments are exploited. The first experiment compares the effectiveness of the two RS at predicting consumer preferences based on a database of explicit ratings of products. The second experiment compares the effectiveness of the two RS at producing Top-N lists based on a real-life customer purchase database. The results of this experiment with Latent Semantic Indexing (LSI/SVD) on two test data sets MovieLens and customer-product purchase data from a large E-commerce company. The metrics like predictions, coverage, statistical accuracy, and Decision support accuracy, Mean Absolute Error (MAE), Root Mean Squared Error (RMSE) and Correlation between ratings are computed. CF based on regression with ratings of active user for items [18]. The task is to predict the preferences (by numerical ratings) of an active user for unseen items preferences given by other users. Based on ratings for some of the items, the experts are combined by using statistical methods to predict the user's preferences for the remaining items. The method was intended to efficiently address the problem of data sparsity and prediction latency. Experiments on movie dataset using Jester benchmark CF data show that this approach achieves improved accuracy than the neighbor based. Further benefits were observed in predicting items with large rating variability on data over an extensive range of sparsity scenarios. In the Netflix data set, the CF based on Restricted Boltzmann Machines (RBM) [16] show that the error rate is (6% over) high than SVD. Likewise, the Probabilistic Matrix Factorization (PMF) [17] is applied and the error rate is (7% over) slightly high than RBM models. A RS based on Matrix Factorization Technique (MFT) [14] is an example for latent factor technique. The strength of MFT is it allows explicit feedback like user score, thumbs up and down called as rating. And implicit feedback such as history, search patterns, and mouse movements. A Non-parametric Matrix Factorization (NMF) is also applied on EachMovie and Netflix dataset [11]. NMF produced more accurate predictions than the traditional low-rank matrix factorization methods of latent factors as SVD, and probabilistic principal component analysis. Restricted Boltzmann Machines Collaborative Filtering (RBM-CF) with non-IID Framework [7] inflates the correlation between user and item ratings.[5] extended the above work and outpaced by probabilistic graphical model and deep belief network for hybrid content-based music recommendation system. In [3], IBCF is related with item-based filtering based on KNN and UBCF. The RS provides high quality recommendations, performing many other recommendations per second for millions of users and items. It attains high coverage in the face of data sparsity. In traditional CF systems, the amount of work increases with the number of participants in the system. Here Item-based techniques first analyze user-item matrix to identify relationships between different items, and then use these relationships to indirectly compute recommendations for users. Different item- based recommendation generation algorithms are assessed for figuring item-item similarities and diverse techniques for obtaining recommendations from them. Finally, results are compared to the basic KNN approach to show that item-based algorithms provide better performance and quality than user-based algorithms.A Domain-sensitive Recommendation (DsRec) for predicting the rating of user-item subgroups with three components like matrix factorization, bi-clustering model and regression regularization is put forth by author [15]. A recommendation considering deep learning [6]. Here, items and users are represented over one-hot encoding of their ID. This utilizes ID data initially designing of the model, which makes a lot of earlier data incapable to be utilized. The effectiveness of feature learning is difficult to ensure. The hybrid model of combining CF with deep learning algorithm in [21]. Initially a quadric polynomial regression model was used for feature representation method then predicting the scores rates. Finally, three datasets were compared with the method which produced better outcome effectively. A Deep Hybrid Recommender System by combining Auto-encoder and Neural Collaborative filtering (DHA-RS)[20] for predicting the preferences from the user-item features. A two-stage recommender system [4] based on CF to predict student grading in combination with faculty ratings to recommend courses for graduate students. Karl Pearson correlation and Cosine Similarity were used to analyze the affinity while CF in the form of cluster approach based on the Artificial Immune Network (AIN) theory is used to identify the similar learner. The dataset includes of both faculties and student though the ratings from faculty do not intervene with predictions. Collaborative Deep Learning (CDL) [12] uses a hierarchical Bayesian model, where deep learning is also performed for user content information. CF is done for the feedback or rating matrix. The Collaborative Topic Regression (CTR) is a methodology that couples two components which learn from two unique sources of information. The results though show improvements, but the hidden representation of CTR will be futile at the times once secondary information is sparse. A progressive Bayesian model using deep learning to get content characteristics and pretty much make recommendations by learning. It also uses a conventional CF model to tackle rating data. But these are not pertinent when we cannot get the content of things. Similarly, [1] was determined to improve Latent Dirichlet Allocation (LDA) based CF by introducing a Collaborative Filtering Based Recommender System (CFBRS). The system works by integrating reviews as a way of representing the ensuing product regularization. They were moved by the success of the LDA based CF approach and the review models based on Neural Network (NN) and Recurrent Neural Network (RNN) to study their effect on CF. NN has the ability to attain resulting product regularization but the RNN seems to be declining the models ability to act regularize the resultant product. Collaborative RS captures user behavior efficiently even when user historical usage is high, and the range of the content remains static. On the other hand, in web application the content may undergo frequent changes even if the user's historical records are either not available or not adequate for making recommendations. This resulted in cold start problem concluding that collaborative recommendation technique unproductive. CBRS attempt to unravel recommendation problem by searching the items for a new user depending on the interest of past users. This means that the item preferences of one user should correlate with the same item of another user. The recommendation framework for e-learning systems [8] based on excellent learners' ratings that indicate the quality of learning materials while recommending the learning materials of similar content. The authors tried to address the problem of growing e-learning material and recommend the material for a specific topic. The system was based on the concepts of Social Learning (SL) and Peer Learning (PL) theories which encourage persuade learners to learn among themselves. This content-based RS has shown noteworthy progress in test results in e-learning as compared to the e-learning systems without recommendation feature. But again, learner rating deviations will affect the result which results in a lower accuracy rate. A novel Content Based recommendation approach [19] which use a network of multiple attribute system for reflecting several other attributes much effectively during the calculation of correlations among the items while recommending them. To ensure that distinct items should be recommended to the users, they employ clustering with the mechanism of centrality, to check interdependence between items and figure out their interaction pattern structure. The method mitigates the problems of overspecialization and sparsity and even claims to overcome the cold start problem by using the past data of the users. The overall scenario related to content-based RS's is that they are simple to implement but, in some situations, it is not enough to seize the exact user preferences using information stored in the user profile. First, the metadata has not been specified entirely in the e-learning systems where there is no service for recommendation feature which leads to the incorrect user profile. Second, the representation of user profiling cannot efficiently capture the relationship among the user items that have been accessed before. Additionally, CBRS are not readily susceptible to the changes adopted by the user, and thus the content filtering system will undergo new user and latest item problem. In [32] context- aware recommendation (CAR) model uses NCF approach and learn nonlinear interactions between latent features of users, items, and contexts. This considers the sequential latent context representation as part of the recommendation process. Since adding of context may increase both the dimensionality and sparsity of the model. A long short-term memory (LSTM) encoder-decoder network on sequences of contextual information (on two context-aware datasets with diverse context dimensions) and extract sequential latent context from the hidden layer of the network to signify a compressed illustration of sequential data.

2.1 Content Aware Collaborative Filtering

The preferences of learners in general shift from context to context in recommender system of E-Learning. Almost in both modes of recommendation techniques context based, and collaborative filtering they take into consideration of only the entries like users and items. They never consider of context information during the recommendation process [22] and [23]. Over to users and items details, the contextual data of users aid to enhance the accuracy and activating the system to furnish the relevant resources exactly. Contextual information is any matter such as knowledge level of user, learning goals or interest level of user etc. to yield recommendations for target audience. Knowledge Level Value (KLV) is taken into consideration as context in the system for recommendation of learning materials. From time to time the knowledge level of user changes. We can find three types of knowledge level value of users such as Beginner (B), Intermediate (I) and Master (M). For the beginner, the learning preferences of resources depend on the knowledge level of user. Over the time, the intermediate learner's knowledge level changes and he chooses different learning preferences of resources. Such variation in preferences of resources is dealt by context aware collaborative filtering. The context information [24] of users added into user and item information to boosting the genuine quality of recommender system. The representation of contextual information as in equation (1), Here item and user are context information added to precisely predict the rating.

$$R: user*item*context \rightarrow Rating \qquad (1)$$

2.2 Neural Collaborative Filtering

Neural Collaborative Filtering (NCF) [6] replicas user and item interactions in the nearest vector space handles proficiently in Neural Network. The traditional approach to solve the RS problem is Matrix Factorization. This bifurcates the user and item matrix called as Utility matrix. This is divided into two sub matrices such as user matrix and item matrix. In the part of prediction process the multiplication of those two sub matrices to rebuild a new utility matrix, in which the larger value more likely that the respective user is interacts with the corresponding item. The utility matrix is factorized in a way such that the loss between the reconstructed matrix permit the true utility matrix getting minimized and a square error. The similarity between user and item latent vector is achieved by using a dot product for each of the latent vectors. NCF contains 4 layers as in figure 4, namely input layer, embedding layer, NCF layer and output layer. The user and item data given to input layer which form the user latent vector and item latent vector in embedding layer, NCF layer contains many layers, finally the concatenated output pass to the output layer.

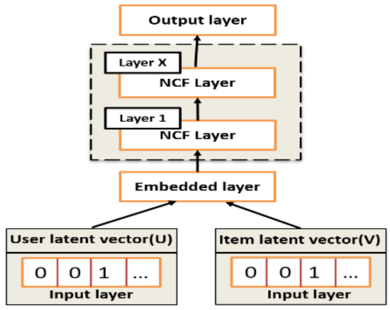


Figure 4. Neural Collaborative Filtering

3. Proposed Recommender System- Context-Aware Neural collaborative Filtering (CA-NCF).

Novel RS technologies are required that can rapidly yield high quality recommendations, for very large-scale problems. To address these issues, we have explored NCF techniques. The proposed hybrid recommender system, CA-NCF uses the contextual information of USERS and materials ITEMS as input. Then similarity between users and items yield similarity matrix, identify neighborhood, predict the rating for unrated items by all the users and finally provide the recommended materials for each user based on their need. "BookCrossing" dataset containing ratings given by users for books is used for implementation. It comprises of three tables and their respective fields like users with user-ID field, books with ISBN field, and ratings with Book-Rating field [26]. BX-Users table contains details about the users such as user-ID, location and age of 278,858 users, BX-Books table covers 271,379 book's information such as ISBN, title of the book, author of the book, year of publication and name of the publisher and BX-Book-Rating table contains 1,149,780 ratings(implicit or explicit) information about book. Rating of the books has minimum value as zero and maximum value as ten [27]. IBCF and UBCF are also applied in the same dataset. For each book, IBCF checks the similarity in terms of similar ratings provided by similar users. It verifies k most related books. For each user IBCF checks most related books for the user's perspective. In other hand, UBCF recognized similar users then suggest top-rated related books.

The architecture of CA-NCF is in figure 5 and the steps of RS process are:

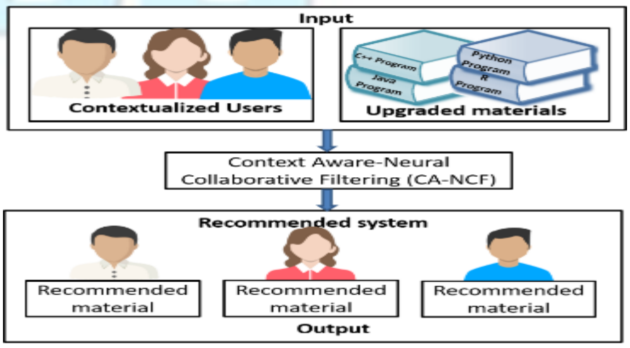


Figure 5. Architecture diagram of the proposed model

i. ***Similarity Computation between Contextualized users/ items:*** Initially, calculation of similarity between the contextualized user and materials using Pearson correlation coefficient is done.

$$\mathrm{Sim}(C_l, C_u) = \frac{\sum_{a=1}^{m}(R_{l,a} - \bar{R}_l)(R_{u,a} - \bar{R}_u)}{\sqrt{\sum_{a=1}^{m}(R_{l,a} - \bar{R}_l)^2}\sqrt{\sum_{a=1}^{m}(R_{u,a} - \bar{R}_u)^2}} \quad (2)$$

Where, $R_{l,a}$ is the rating given by target learner 'l' to learning material 'a' ,$\bar{R}_l$ is mean (average) rating of all the ratings given by target learner 'l' on contextual information of learners. Similarly, $R_{u,a}$ signifies the rating by learner 'u' to learning material 'b' and $\bar{R}_u$ is average rating of all the ratings provided by learner 'u.' The total number of contextual information denoted as 'M' is used to compute the ratings and mean rating. It is based on KNN approach.

ii. ***Formation of Neighborhood:*** In the next step, the neighborhood is formed based on similarity learner (that is high value is most similar) from equation (2).

iii. ***Prediction of Ratings:*** Finally, prediction of ratings is acquired based on equation (3)[25].

$$P_{l,b} = \bar{R}_l + \frac{\sum_{u=1}^{n}(R_{u,b} - \bar{R}_u) \times Sim(C_l, C_u)}{\sum_{u=1}^{n} Sim(C_l, C_u)} \quad (3)$$

Here, $P_{l,b}$ is the prediction rating for the learning resource b by target learners l, n is the total number of neighbors in the neighborhood formed by contextual similarity computation.

iv. ***Contextualized Recommendations of materials***
The learning materials are predicted depending upon the ratings of each item by similar interest users into the perspective to achieve the top recommendation for the user by personalized services.

4. Implementation of Context-Aware Neural collaborative Filtering (CA-NCF).

Intended for the implementation "recommenderlab" package in R programming is included. It contains functions to implement the recommendation technique. Methods from "recommenderlab" are:

- **similarity()** - To compute the similarity amid users/items with Pearson method.
- **predicts()** - To find the rating value for unrated items.
- **normalize()** - To normalize the data.
- **evaluate()** - To evaluate performance.
- **evaluationScheme()**-Bootstrap is the alternate technique to split the data.
- **recommender()** - To generate top N recommendations for IBCF, UBCF and CA-NCF.
- **plot()** - To display the precision, recall, and ROC Curve of the generated model.
- **getConfusionMatrix()** - To create the confusion matrix value for recommended learning resources.

Algorithm 1 defines the method of generating contextualized recommendations of materials based on user preferences and contextual values using CA-NCF algorithm.

Algorithm 1: Generate Contextualized Recommendation
Declare Users x={$x_1, x_2 .. x_n$},
Items a={$a_1, a_2 .. a_m$}
Context C = {1, 2, 3}
Ratings R = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}
for (i=1; i<=n; i++)
Compute similarity sim(C_x, C_y) using equation (2) based on contextual.
Predict ratings $P_{x,b}$ for unrated material b by target learner x using equation (3).
Generate top N recommendations by CA-NCF algorithm.

5. Results and performance discussion

Confusion matrix for RS for recommended and not recommended materials, relevant and not relevant materials as shown in table 1. Where,

- RR- Relevant Resources Recommended.
- RN- Relevant Resources Not Recommended.
- NR- Not Relevant Resources Recommended.
- NN- Not Relevant Resources Not Recommended.

Table 1 Confusion Matrix for Recommender system

	Recommended	**Not Recommended**
Relevant	RR	RN
Not Relevant	NR	NN

To evaluate top-N recommendation we opted two metrics that are widely used in the information retrieval community namely recall and precision. Other evaluations calculated for these algorithms are accuracy and F1 measure [31].

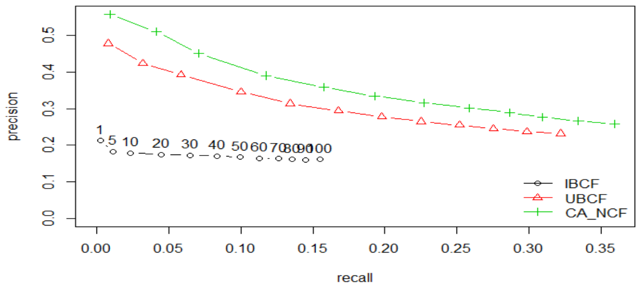
Precision is rating of learning materials recommended to the total number of learning materials.

$$\boldsymbol{Precision} = \frac{\textit{Recommended Learning Resourses}}{\textit{Total Learning Resourses}} = \frac{RR}{RR+NR} \quad \textbf{(4)}$$

Recall is the ratio of correctly recommended learning materials to the relevant learning materials.

$$\boldsymbol{Recall} = \frac{\textit{Correctly Recommended Learning Resourses}}{\textit{Relevant Learning Resourses}} = \frac{RR}{RR+RN} \quad \textbf{(5)}$$

The evaluation of precision and recall is shown in figure 6. The proposed CA-NCF has high precision and low recall, shows the good precision compared to IBCF and UBCF. The precision and recall values are calculated for the points such as 1, 5, 10, 20, 30, 40, 50, 60, 70, 80 90, and 100. For CA-



NCF when the recall is 0.36, 0.01 then the precision is 0.25, 0.6 respectively. For UBCF when the recall is 0.32, 0.01 then the precision is 0.24, 0.48. For IBCF when the recall is 0.16, 0.01 then the precision is 0.1, 0.2 respectively. The CA-NCF yields high precision 0.6 and low recall 0.01 when compared to other recommendation techniques.

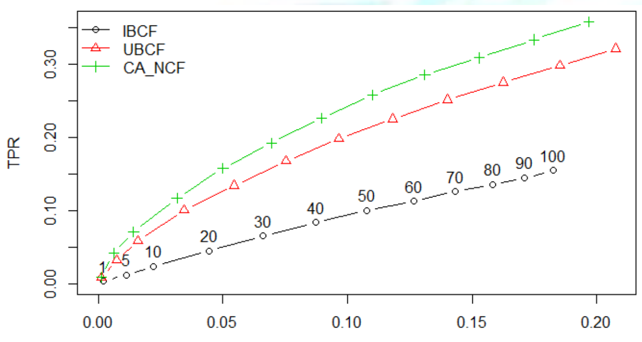


Figure 6. Precision and recall graph

Figure 7 shows the assessment of Receiver Operating Characteristic (ROC) Curve with True Positive Rate (TPR) and False Positive Rate (FPR). CA-NCF has the high TPR and low FPR value to display good precision compared to other techniques. The FPR and TPR values are calculated for the points such as 1, 5, 10, 20, 30, 40, 50, 60, 70, 80 90, and 100. For IBCF when the FPR is 0.00, 0.19 then TPR is 0.00, 0.15 respectively. For UBCF when the FPR is 0.00, 0.20 then the TPR is 0.00, 0.30 respectively. For CA-NCF when the FPR is 0.00, 0.19 then the TPR is 0.00, 0.35 respectively.

Figure 7. ROC Curve

Conclusion

Recommender system is used to predict and suggest the required learning resources for the learners. The context information is considered for the recommendation system to predict the knowledge level of learners whether they belong to beginner, intermediate or master. These context details are added to both user and item dataset. Thus, accuracy of the system is improved. The performance of the recommender system is shown superior of fabricating high accuracy through evaluation metrics like precision, recall and Receiver Operating Characteristic curve with respect to confusion matrix value of True Positive Rate and False Positive Rate.

References

[1] Almahairi, Amjad, Kyle Kastner, Kyunghyun Cho, Aaron Courville, A. (2015,). Learning distributed representations from reviews for collaborative filtering. In Proceedings of the 9th ACM Conference on Recommender Systems (pp. 147-154). ACM.
[2] Badrul M. Sarwar, George Karypis, Joseph A. Konstan, John T. Riedl. (2000). Application of dimensionality reduction in recommender system-a case study (No. TR-00-043). Minnesota Univ Minneapolis Dept of Computer Science.
[3] Badrul M. Sarwar, George Karypis, Joseph A. Konstan, John T. Riedl. (2001). Item-based collaborative filtering recommendation algorithms. World Wide Web, Vol. 1, pp. 285-295.
[4] Chang, Pei-Chann, Cheng-Hui Lin, Meng-Hui Chen. (2016). A hybrid course recommendation system by integrating collaborative filtering and artificial immune systems. Algorithms, Vol. 9(3), 47.
[5] Wang, Xinxi, Ye Wang. (2014). Improving content-based and hybrid music recommendation using deep learning. In Proceedings of the 22nd ACM international conference on Multimedia (pp. 627-636). ACM.
[6] He, Xiangnan, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, Tat-Seng Chua. (2017). Neural collaborative filtering. In Proceedings of the 26th international conference on world wide web (pp. 173-182).
[7] Georgiev, Kostadin, Preslav Nakov. (2013). A non-iid framework for collaborative filtering with restricted boltzmann machines. In International conference on machine learning (pp. 1148-1156).
[8] Ghauth, Khairil Imran, Nor Aniza Abdullah. (2010). Learning materials recommendation using good learners' ratings and content-based filtering. Educational technology research and development, 58(6), 711-727.
[9] Han, Ji-won, Jae-choon Jo, Hye-sung Ji, Heui-seok Lim. (2016). A collaborative recommender system for learning courses considering the relevance of a learner's learning skills. Cluster Computing, Vol. 19(4), pp. 2273-2284.
[10] Jung, Kyung-Yong, Dong-Hyun Park, Jung-Hyun Lee. (2004). Hybrid collaborative filtering and content-based filtering for improved recommender system. In International Conference on Computational Science (pp. 295-302). Springer, Berlin, Heidelberg.
[11] Kai Yu, Shenghuo Zhu, John Lafferty, Yihong Gong. (2009). Fast nonparametric matrix factorization for large-scale collaborative filtering. 32nd international ACM SIGIR conference on Research and development in information retrieval (pp. 211-218). ACM.
[12] Hao Wang, Naiyan Wang, Dit-Yan Yeung. (2015). Collaborative deep learning for recommender systems. 21th ACM SIGKDD international conference on knowledge discovery and data mining (pp. 1235-1244). ACM.
[13] Koji Miyahara,Michael J. Pazzani. (2000). Collaborative filtering with the simple Bayesian classifier. In Pacific Rim International conference on artificial intelligence (pp. 679-689). Springer, Berlin, Heidelberg.
[14] Koren, Yehuda, Robert Bell, Chris Volinsky.(2009). Matrix factorization techniques for recommender systems. Computer, Vol. (8), pp. 30-37.
[15] Liu, Jing, Yu Jiang, Zechao Li, Xi Zhang, Hanqing Lu. (2015). Domain-sensitive recommendation with user-item subgroup analysis. IEEE Transactions on Knowledge and Data Engineering, Vol. 28(4), pp. 939-950.
[16] Ruslan Salakhutdinov, Andriy Minih Geoffrey Hinton. (2007). Restricted Boltzmann machines for collaborative filtering. In 24th international conference on Machine learning (pp. 791-798). ACM.
[17] Ruslan Salakhutdinov,Andriy Mnih. (2008). Probabilistic matrix factorization. In Advances in neural information processing systems (pp. 1257-1264).
[18] Slobodan Vucetic, Zoran Obradovic. (2005). Collaborative filtering using a regression-based approach. Knowledge and Information Systems, Vol. 7(1), pp. 1-22.
[19] Son, Jieun, Seoung Bum Kim. (2017). Content-based filtering for recommendation systems using multiattribute networks. Expert Systems with Applications, 89, 404-412.
[20] Yu Liu, Shuai Wang, M. Shahrukh Khan, Jieyu He. (2018). A novel deep hybrid recommender system based on auto-encoder with neural collaborative filtering. Big Data Mining and Analytics, Vol. 1(3), pp. 211-221.

[21] Zhang, Libo, Tiejian Luo, Fei Zhang, Yanjun Wu. (2018). A recommendation model based on deep neural network. IEEE Access, Vol. 6, pp. 9454-9463.
[22] Zheng, Yong, Bamshad Mobasher, Robin Burke (2015). Similarity-based context-aware recommendation. International Conference on Web Information Systems Engineering (pp. 431-447). Springer, Cham.
[23] Adomavicius, Gediminas, Alexander Tuzhilin. (2011). Context-aware recommender systems. In Recommender systems handbook (pp. 217-253). Springer, Boston, MA.
[24] de Campos, L. M., Fernández-Luna, J. M., Huete, J. F.,Rueda-Morales, M. A. (2010). Using second-hand information in collaborative recommender systems. Soft Computing, Vol. 14(8), pp. 785-798.
[25] Jannach, D., Zanker, M., Felfernig, A., & Friedrich, G. (2011). Recommender systems, an introduction. Hardback.
[26] Zaier, Zied, Robert Godin, and Luc Faucher. (2008, November). Evaluating recommender systems. In 2008 International Conference on Automated Solutions for Cross Media Content and Multi-Channel Distribution (pp. 211-217). IEEE.
[27] Ziegler, Cai-Nicolas, McNee, S. M., Konstan, J. A., Lausen, G. (2005) Improving recommendation lists through topic diversification. In Proceedings of the 14th international conference on World Wide Web (pp. 22-32). ACM.
[28] Ricci, Francesco, Lior Rokach, Bracha Shapira. (2011). Introduction to recommender systems handbook. In Recommender systems handbook (pp. 1-35). Springer, Boston, MA.
[29] Erdt, Mojisola, Alejandro Fernandez, Christoph Rensing. (2015). Evaluating recommender systems for technology enhanced learning: a quantitative survey. IEEE Transactions on Learning Technologies, Vol. 8(4), pp. 326-344.
[30] Pazzani, Michael J.,Daniel Billsus (2007),"Content- based recommendation systems. In The adaptive web (pp. 325-341). Springer, Berlin, Heidelberg.
[31] Manning, Christopher, Prabhakar Raghavan, Hinrich Schütze. (2010). Introduction to information retrieval. Natural Language Engineering, Vol. 16(1), pp. 100-103.
[32] Livne, A., Unger, M., Shapira, B., Rokach, L. (2019). Deep Context-Aware Recommender System Utilizing Sequential Latent Context. *ArXiv.*