IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-7, Special Issue, Feb 2019 E-ISSN: 2347-2693

A Novel Recommender System based on Artificial Neural Network Learning Vector Quantization Classification Approach

S. Prasanna Priya[1*], M. Karthikeyan[2]

[1]Administrative Office, Annamalai University, Annamalinagar, *Tamilnadu.*
[2] Department of Computer and Information Science, Annamalai University, Annamalinagar, *Tamilnadu*

Corresponding Author: monishridevanathan@gmail.com

Available online at: www.ijcseonline.org

Abstract— Recommender systems have become more important in various domains for lessening the issue of information overload. Traditional Recommender Systems are Collaborative filtering method and Content based filtering method.However, these recommendation methods suffer from data sparsity and cold start problem. So this paper proposes an ANN based recommender system. Artificial Neural Network –Learning Vector Quantization (ANNLVQ) and Optimized Learning Vector Quantization (ANNOLVQ) algorithms are used todevelop a multi-categorical classification model that predicts the class of a rating in recommender systems. In this proposed research, the problem of predicting the rating as a multi-label classification problem is considered where each rating has treated a label. Book dataset used for this proposed research. ANN recommender systems accuracy compared with collaborative filtering method recommender system and ANN recommender systems predicts more accuracy than traditional collaborative filtering method.

Keywords— *Artificial Neural Network, collaborative filtering, Learning Vector Quantization, Book Recommendation, Recommender systems.*

I. INTRODUCTION

In the era of information overload, Internet users may find it difficult to choose from the multitude of available products and services. There is a requirement aimed at Recommender systems (RSs) that create modified recommendations [1]. The idea behind RSs is not new. It is general to enquire associates for references when one selects an eatery, film, book, etc. To make a recommendation, an RS usually needs user data, items, and user feedback on those items. Subsequently generating a recommendation, user response on the item is acquired either openly or indirectly [2].

The utilization of RSs in e-commerce has many benefits for both sellers and consumers. The former's objective is to make their products available to concerned clients and to achieve consumer satisfaction in addition to loyalty. Their objective can be achieved when users regularly receive products that meet their needs [3]. On the other hand, consumers would receive a list of products they would be most likely to find useful. They also save time, effort and money they would spend trying to discover items they truly appreciate [4].

Recommender systems are useful for internet users who may find it hard to choose from the multitude of available products and services. RSs predict how likely the target user is to be interested in an item which might have been unknown to them [5]. This study deliberates book recommender systems that could be beneficial in the public library, schools, and on e-learning portals. Currently, with the introduction of e-books, readers can access low-cost resources using less effort. It was estimated that the performance of reading for desire would turn out to be prevalent, but statistics prove the opposite; it is declining, mainly between young people.

Collaborative filtering (CF) methods most often consider the user as a static entity whose interests are fixed in time. Matrix factorization, for instance, utilizes all the ratings (or implicit response) of a user to build a demonstration of its familiar tastes, ignorant to the probable evolution of taste or fading interests of the user [6]. To make recommendations, CF only requires an item-user rating matrix, so it is simple to develop. The rating matrix, however, can be sparse, specifically in the instance of new items or new users; this is named as cold startproblem. Sparsity may advance to poor recommendations. CF has two other drawbacks. The gray sheepissue happens when the system calculates the preferences of a user who has a task dissimilar than other users. The shilling attackissue happens once an item accepts false evaluations as a form of promotion [7].

A CB recommender is a classifier that studies the patterns in addition to similarities in the buying history of a user to forecast her future interests. A book's content could check its title, summary, outline, whole text or metadata, containing author, year of publication, publisher, genre, page number, etc. [8].Existing techniques for recommender systems [9-13] mainly categorized into collaborative filtering and content-based recommendations. Collaborative filtering depends heavily on user activities, e.g., ratings of items according to their preferences. The recommender functions under the assumption of similar preferences among users and a sufficient number of user ratings available in the system.

Collaborative filtering, however, has difficulty handling items without sufficient numbers of user ratings and new items that one has purchased or rated, i.e., the so-called cold start problem. In particular, most books are seldom utilized by many patrons according to library use statistics. Thus, effective content-based recommendations become important when these user activities are sparse. A content-based method has been initially developed for book recommendations. Its system, however, depends on a careful feature selection process by labeling every book with values, which is a labor-intensive task. Specific attributes of users must also be provided in advance when evaluating recommended books. Automated text-classification approaches then employed for exploring content-based book recommender systems. However, the relevance of the recommendations only considered textual metadata, partially extracted from the Internet, rather than actual book text. In industrial applications, e.g., Google Books, full-text indexing has been used commonly for book retrieval via search queries.

In a recommendation systems context, metadata information from analyses inscribed for businesses has infrequently deliberated in traditional systems technologically advanced with content-based in addition to collaborative filtering methodologies [14]. Collaborative filtering, as well as content-based filtering, are common memory-based techniques for endorsing innovative products to the users but grieve from certain drawbacks and be unsuccessful to offer practical recommendations in various circumstances. The sparsity mentions to the enormous percentage of nonappearance of ratings: individually user only has ratings in a very restricted number of the accessible data. There is no sufficient space for complete data [15].

This paper proposes a Classification Approach to develop a multi-categorical classification model that predicts the class of a rating using Artificial Neural Network LVQ and OLVQin recommender systems and toenhance prediction accuracy. In this proposed approach, the problem of predicting the rating as a multi-label classification problem considered where each rating has treated a label. Collaborative Filtering Cosine Similarity (CFCS) recommender system compared with ANN recommender systems for accuracy.

This paper is structured as: Section 2 explains the literature review related to recommendation systems; Section 3 defines the proposed methodology; Section 4 gives the results and discussions followed by the conclusion in addition to future scope in section 5.

II. LITERATURE REVIEW

Suglia et al. (2017) [16] presented a modular and extensible architecture exploiting deep neural networks to provide users with content-based recommendations. The model was based on LSTM networks, a particular class of RNNs particularly able to deal with sequences of data of arbitrary length, as the content describing the recommended items. Specifically, an approach was developed which jointly learns two embeddings on behalf of both items to be recommended as well as user's preferences. Given such representations, recommendations were provided by exploiting a logistic regression layer which calculates the likelihood that a user will like a particular item. The results of the proposed approach showed that the proposed deep architecture was able to significantly overcome both algorithms based on (shallow) neural networks as Word2Vec W2V and Doc2Vec D2V as well as popular and well-performing techniques for collaborative filtering and matrix factorization.

Devooght and Bersini (2017) [17] showed that recurrent neural networks are a powerful tool aimed at collaborative filtering, even external to the sparse session-based settings where it first introduced. This method achieved the best results using the categorical cross-entropy objective function. RNNs performed exceptionally well on short-term recommendations and adding noise to the training sequence was observed (such as dropout and shuffling) improves its triumph on long-term recommendations.

Yi et al. (2016) [18] suggested an expanded auto encoder recommendation framework Supervised Neural Recommendation (SNR). The stacked auto encoders model was considered to excerpt input feature formerly reconstruction of the input to make the recommendation. Then the side information of objects was mixed in the structure, and the Huber function based regularization was implemented to enhance the performance of recommendation. The first innovation of current recommendation framework was that the side information was used to enlarge the framework. The presented scheme was verified on a public dataset. Results indicated that the recommendation framework has better performance than the state-of-art recommendation methods.

Veugen, T., & Erkin, Z. (2015) [19] developed a system for recommending items in a privacy-preserving way by using a content-based item similarity matrix. Compared to previous solutions, the leakage of the divisors Vi was avoided which contain information about the commercially sensitive item similarities. The costs of introducing a secure division protocol led to a doubling of the computational and communication complexity, and a slight loss in recommendation accuracy. However, the system neither relied on trusted third parties nor requires interaction with peer users. In addition, this proposal offered an efficient and much more secure solution for this class of recommender systems.

Shen et al. (2016) [20] suggested a novel CNN to make personalized recommendations and attained a superior result. The suggested procedure tested on a public dataset. Outcomes specified in this recommendation procedure for recommending new and unpopular learning resources was practicable. The CNN-based model would show a vital role in e-learning systems or intelligent tutoring systems. Even though the application considered here was learning resources recommendation, the technique was more usually appropriate to news recommendation, and so on.

Chen, L., & Wang, F. (2017) [21] presented a method of implementing tradeoff-oriented explanations in preference-based recommender systems. Through measuring users' objective behavior and subjective perceptions as well as collecting their free comments in both before-after and within-subject's experiments, several interesting findings: 1) Incorporating feature sentiments into Pref-ORG can be effective to increase users' product knowledge, preference certainty, perceived information effectiveness, recommendation clearness, and quality, and buying intention. 2) The explanation interface's actual effectiveness was also measured, which indicates almost half of users made better choices after using SentiORG. 3) As for decision efficiency, it shows users spent more time in making decisions in Senti-ORG, which is consistent with related works of literature' observation that efficiency is not necessarily correlated to users' decision effectiveness and perceived system competence. 4) Three design principles derived from the experiment results. In particular, given that The majority of users preferred mixture View, recommended explaining products' tradeoff properties (pros and cons) regarding both feature sentiments and static specifications.

Paradarami (2017) [22] technologically advanced an innovative hybrid RS procedure that forms on the capabilities provided through traditional methodologies similar to collaborative filtering besides content-based filtering by employing the metadata related with review text to train and build an Artificial Neural Network (ANN). A multi-categorical classification model was established that forecasts the class of a rating. LogLoss, a convex function, was the cost function reduced by relating stochastic gradient descent and prediction of accuracy was utilized to determine the model's efficacy. The effectiveness of the hybrid model assessed by analyzing the percentage of observations with correct predictions. The efficacy of these rating predictions also assessed when translated to yes/no recommendations.

Tewari, A. S., & Barman, A. G. (2018) [23] stated that almost all existing e-commerce recommendation systems had put all their efforts in augmenting all exciting items of the target user to their recommendation list, deprived of any importance to the order of things in the recommendation list. The suggested method meaningfully has exposed around 34% precision for top-n recommendations. The proposed approach had its unique feature that finds the popularity of items in the market using opinion mining. All these unique features collectively help the proposed RS in creating relevant smaller topn recommendations list for the target user and also assist in alleviating item side cold start and gray sheep problems. The investigational outcomes presented that the suggested RS significantly outperformed the further benchmark recommendation techniques.

Liu et al. (2018) [24] proposed an online activity recommendation approach based on the dynamic adjustment of a recommendation list be implemented on NiusNews, an online news website. User preferences were derived by studying the latent issues founded on Non-Negative Matrix Factorization (NMF) and the hidden topics based on Latent Dirichlet Allocation (LDA). The concerns of sparse data and cold-start activities were alleviated by carrying out a possible association study of news in addition to activities. Furthermore, the current news was considered that the target consumers were looking for capturing the current preferences more precisely. To manage the concern of limited recommendation layouts, the Most Frequently Pushed (MFP) and Not Frequently Clicked (NFC) replacement approaches were recommended for a dynamic variation of the recommendation list. These strategies are critical for practical purposes of online recommendations and not considered in existing recommendation methods. The proposed replacement approaches in addition to online recommendation method offer probable solutions for dynamically adjusting recommendation lists in online recommender systems. The developed system was dynamically adaptive to cost and efficiency. The suggested approaches (FAR-ONHI and WAR-ONHI) integrate user preference study and existing news interest study through the activity replacement strategy to dynamically adjust the recommendation lists. In the online experiment, the results showed that the proposed method performed better than other methods do. The online evaluations demonstrated that the proposed approach considering the dynamic adjustment of recommendation lists could improve recommendation quality for online recommendations.

III. PROPOSED METHODOLOGY

The ANNLVQ and ANNOLVQ approach is introduced in this section, starting with the intuition behind it, and then continue to describe its details. While the proposed method is general and can be used to recommend any consumer item, a specific example domain for illustration is chosen. Providing the most considered consumer domain, in the background of recommendation, is that of books, henceforth, usage of the book domain to present the proposed technique. In other words, the suggested technique is present to recommend books founded on customer reviews.

Intuition and overview

The overall goal of RSs is to select the objects that could be of concern to a user. In this proposed context, predicting ratings for books to users, and recommending those books with highest expected ratings to them, together with reasonable and personalized explanations to improve the transparency of the logic in the recommendations is presented. Mentioning that the number of descriptive attributes that typically utilized in content-based book recommendation is limited and inadequate, the proposed approach spontaneously extract adjective features from external user reviews to describe individual features of items besides user tastes that are capable to truthfully reflect the users' perception towards books at a higher and more abstract level.

The proposed technique states the rating sparsity problem by decomposing a singular user rating into multiple measurements considered by extracted adjectives and then converting a minor number of user ratings into a more significant number of feature preferences. This permits to comprehend user benefits well, and to choose their selected items more precisely through each of their preferred features, consequently alleviating the issue of item-level rating sparsity. Furthermore, by explicitly listing out adjective features that cause items to be endorsed, the user could be competent to clarify the reason for the recommendations instinctively to users, with the objective of addressing the transparency problem.

Recommendation framework

The overview of the suggested book recommendation framework is presented in Fig.1. The main components are designed and implemented to realize the proposed recommendation engine; they are Data Collection, Pre-Processing, Feature Extraction , Recommendation using Collaborative filtering method, Recommendation system using ANNLVQ classification , Recommendation using ANNQLVQ and Accuracy prediction.

Dataset Collection

The dataset used for the proposed method is collected from the database which has the details of the user ratings on various books. The dataset contains both explicit and implicit feedback. Datasets are generated in a .csv file, and it has the details of book_id, best book, work_id, ISBN, Original publication, average rating, rating count, work_rating and work test review from comment.

Pre-processing

Data pre-processing is usually the initial stage of knowledge discovery. Data pre-processing can influence simplification performance of a classification algorithm. Most of the real world datasets suffer from problems of missing values and ambiguities; similar was the case with the proposed dataset. So Pre-processing process is must for dataset.

First, all the rows or information on articles which didn't belong to any class are removed. There was some amount of user feedbacks in which article didn't belong to any class at first, after removing all of them were left with around some user feedbacks and classes. Next, ambiguities in the feature 'classes' were removed, ambiguities like several classes were referring to the same author or entity, it is replaced all of them by a single class.

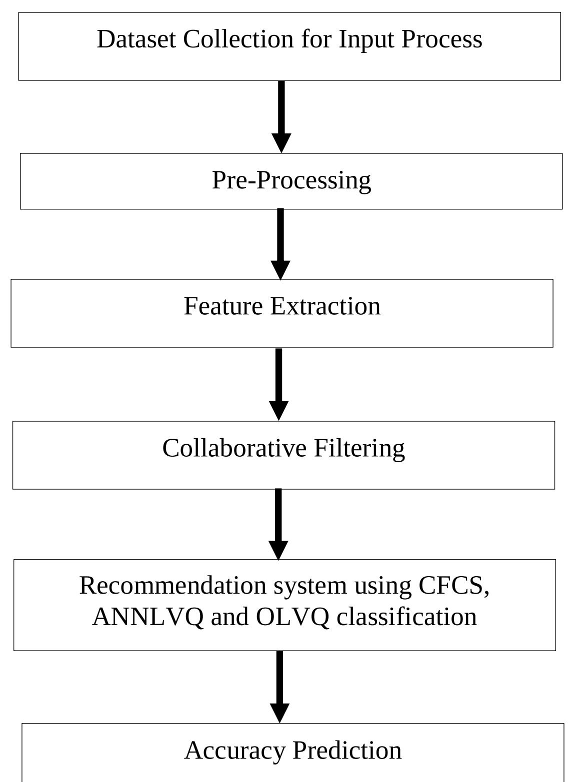

Fig. 1. The framework of the proposed book recommendation system

After the second step, all the classes which occurred only once as classification requires the occurrence of each item was removed at least twice so that one can be used in training and other for testing. After the third step, it is rechecked if there was any article still left which didn't belong to at least one class. The process is stopped when a sufficient amount of ambiguities and each article belonged to at least one class were removed. After performing all the steps above, finally left with a less number of user feedbacks and different minimum classes.

Collaborative Filtering based Recommendation system

To obtain predictions and recommendations to collaborative filtering a subset of active users is chosen to the criteria for selecting a subset is based on the user's similarity with active users then the weighted aggregate is computed for their rating to generate recommendations. Collaborative filtering comprises of the three major steps. In initial step all users will be weighted and similarity is computed with corresponding to the active user. In the second step subsets of user called as predictors will be designated. In the third step rating is normalized and the weights of selected neighbours are combined with rating to make prediction [25-26].

At that point the documents are represented as term vectors, the comparability of the two documents relates to the relationship between the vectors. This is evaluated as the cosine of the point between vectors, that is, the so-called as cosine similarity. Cosine similarity is a standout amongst the most well-known similarity measure employed to text documents.

Input-**User and book ratings**

Output-**similarities between user and books**

$$\text{Similarity} = \cos(\theta) = \frac{A.B}{\|A\|\|A\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} {A_i}^2}.\sqrt{\sum_{i=1}^{n} {B_i}^2}} \quad (1)$$

The process steps are

1. Get the dot product of vectors 'a' and 'b'
2. Multiply magnitude 'a' and magnitude 'b'
3. Divide the dot product of vectors 'a' and 'b' by the product of magnitude 'a' and magnitude 'b'.

Accuracy prediction CFCS,ANNLVQ andANNOLVQ classification

An ANN is a machine learning approach that uses a combination of similar models to improve the outcomes attained over a single model. In this paper, Collaborative Filtering Cosine Similarity- ANN optimized Learnig Vector Quantization Classification approach is used to predict the class of the book based on the reviewer comments.

Vector quantization (VQ) is a common algorithm in the fields of text and speech processing. Having N information vectors, VQ algorithm groups them into a small number of clusters in an unsupervised methodology. VQ might be considered as a clustering technique. Optimized Learning Vector Quantization (OLVQ) is a neural network that joins competitive learning with supervision. It very well may be utilized for pattern classification [27]. Optimized LVQ joins clustering and classification method dependent on feed forward neural network. Input sources are stimulated through a variable number of hidden layers to the output nodes.

Initially, the information space is distributed into non-overlapping regions or clusters. Second these regions are mapped to predefined classes. The initial step has been sophisticated utilizing a competitive layer of the network that works like the Self-Organizing Map (SOM). The layer clusters the information vectors utilizing a table of vector models known as a codebook. The quantity of codebook vectors is substantially less than the quantity of input information vectors. Nonetheless, it must be predefined by the client. In the clustering task, each data vector is allocated to the nearest codebook vector as per a predefined distortion measure. The second step of optimized LVQ is practiced utilizing the linear layer of the system that maps each codebook vector to the object class. The design of Kohonen‟s Neural system that executes optimized LVQ tasks. It consists of three layers; named input, hidden called competitive, and output called linear layers. The weights of the input-competitive links represent the codebook vectors. They are M-dimensional vector, as the input vectors, that are located in the input data space for identifying cluster regions. Clusters borders are defined by a "Voronoï net‟ of hyper-planes perpendicular to the linking line of two codebook vector. Each neuron in the competitive layer represents one's cluster. The linear layer maps the competitive layer's neurons into target classification defined by the user. Multiple neurons may be appropriate to the similar class, however, in the data space, cluster regions equivalent to the similar class in the M-dimensional space need not be contiguous. The learning algorithm has to properly locate the competitive 's neurons, codebook vectors, in the M-dimensional input space and subordinate them to the correct linear neurons, class labels [28].

Learning algorithms are all adaptive as the training samples are presented one at a time in random order. The codebook vectors gradually capture the fundamental statistical properties of the training data. That is for avoiding both the falling in local optima and the difficulty of gradient calculation. As an outcome, optimized LVQ networks are statistical classifiers, which quickly converge to a good solution.

The initial step in the optimized LQV neural network design is the parameters setting of both competitive and linear layers. Then the presented input data vectors have to be separated into training and test groups. Learning algorithm normally works as follows:

Initialization of Codebook:
For each target class, the number of codebook vectors has to be relative to the number of incidence of that class and these vectors are adjusted to the center of the input ranges.

Determination of the winner:
The Euclidean distance has to be evaluated between training data vector and every codebook vector,

$$d = \| W_j - x_i \| = \Sigma_i (W_j - x_i)^2 \quad (2)$$

$$m_c = \arg\min(d_i) \quad (3)$$

Codebook Adaptation:
Codebook vectors are optimized during the learning process. They are all iterative gradient methods. The requirement for finding the optimal codebook and avoid difficult gradient calculations.

Optimized LVQ1 initiates by randomly chooses a training vector x, discovers the nearest codebook vector mc which is called the winner and transfers this winning neuron to the training data vector if both of them belong to the same class, if not the neuron will be moved away and all other neurons are kept unchanged [29].

$$m_c(t+1) = m_c(t) + s(t)\alpha(t)[x(t) - m_c(t)] \quad (4)$$

$$m_i(t) = m_i(t) \text{ for } i \neq c.$$

Based on this the class of the books is classified based on the reviewer comments.

Accuracy Prediction
In accuracy evaluation of classification, there are Recall, Precision and F-measure to evaluate the overall accuracy of the classifier.

Recall
A recall is the fraction of the correctly classified instances for one class of the overall instances in this class. For example, if 900 books are classified to positive and 800 of them are correct, and in the dataset, there are 1000 books which are positive, then the recall for the positive class is 800/1000, which equals to 0.8.

Precision
Precision is the fraction of the correctly classified instances for one class of the overall instances which are classified to this class. For example, if 900 books are classified to positive and 800 of them are correct, and in the dataset, there are 1000 books which are positive, then the Precision for the positive class is 800/900, which equals to 0.89.

F-measure
To get a comprehensive evaluation of the classification, F-measure is developed to integrate the Recall and the Precision. The F-measure can be expressed as

$$F_\beta = (1+\beta^2) * \frac{\text{Precision} * \text{recall}}{\beta^2 * \text{Precision} + \text{recall}}$$

(5)

This is a general form of F-measure, and the parameter β is used to change the weights for Precision and Recall in calculating the F-measure value.

IV. RESULTS AND DISCUSSIONS

The CFCS, ANNLVQ and ANNOLVQ classification based recommender system is simulated in the environment of Java.

Home page and Upload the Dataset:
Fig.2 describes the collecting and uploading of the book recommendation dataset. Here, the dataset is a structured dataset which is a .csv or SQL file.

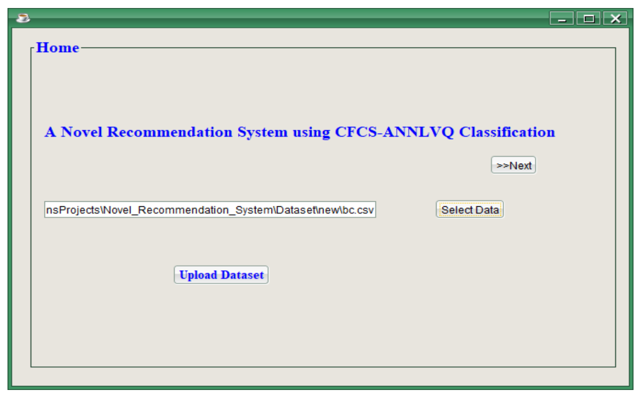

Fig .2. Uploading the dataset

After selecting the data, that particular path will be displayed in the text box. By clicking the uploaded dataset, the response dialog box is open, and it is shown in fig.3 and fig.4.

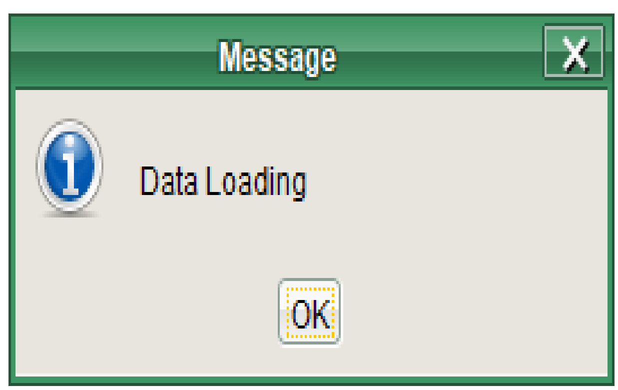

Fig. 3. Data uploading

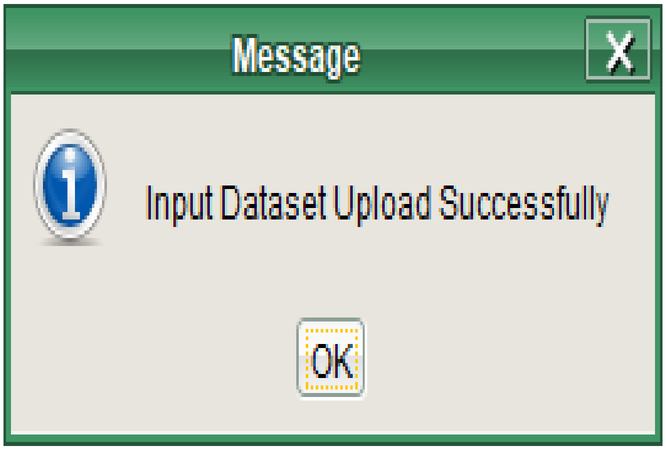

Fig. 4. Successful updating of input dataset

After uploading the data, all data contents will be displayed as shown in fig.5. It will view the uploaded dataset.

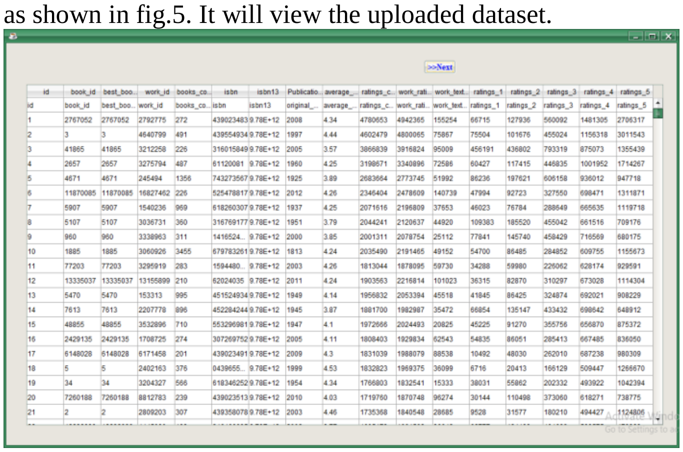

Fig. 5. Displaying of the contents

Pre-processing

In pre-processing, initially, the input datasets are given which consists of 5042 input records. This is given as an input. It has some missing attributes. After pre-processing, missing attributes are eliminated.

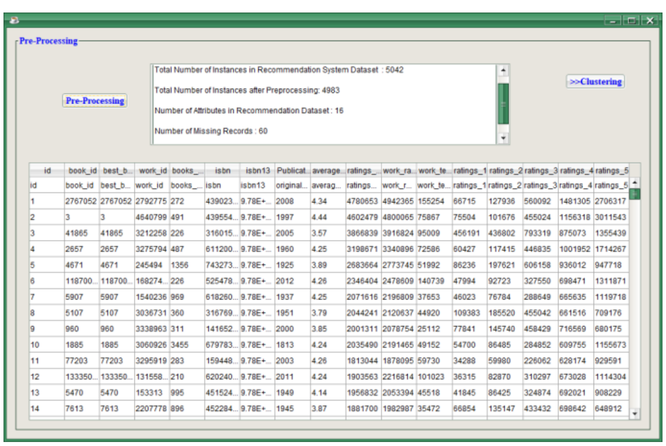

Fig. 6. Pre-processing

As shown in the above fig.6, the total number of instances in the recommendation system dataset is 5042 input records, and the total number of cases after pre-processing is observed as 4983. The number of attributes in recommendation dataset is 16, and the quantity of missing records is 60. The quantity of missing records is given as shown below:

Number of missing records= (number of records before pre-processing- the number of records after pre-processing)

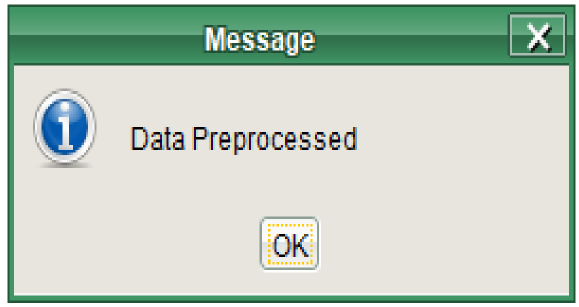

Fig. 7. Data pre-processed

After the process of data pre-processing is completed, then a dialog box is opened as shown in fig.7.

Collaborative Filtering

In the proposed system, Collaboration Filtering for recommendation system through Cosine Similarity matrix as shown in fig.8. Fig.9 shows the book recommendation status of the system using datasets. Fig.10 shows the computation of confusion matrix and Similarity Score By suing the proposed recommendation system 78.359% accuracy predicted through confusion matrix as shown in fig.11.

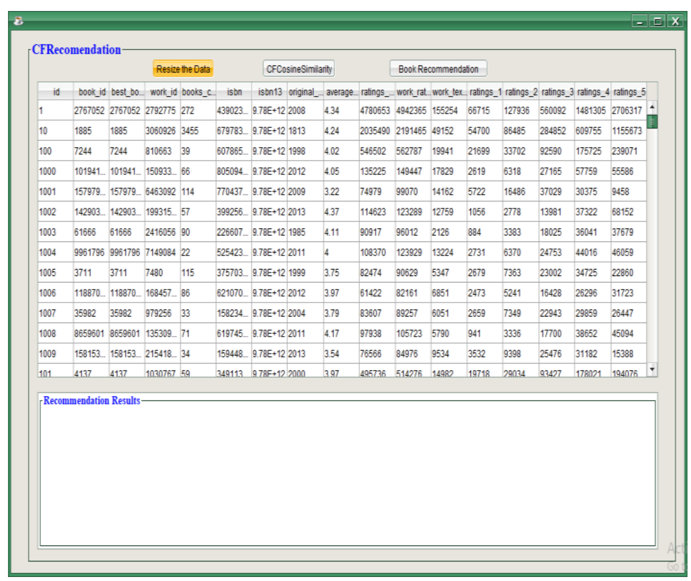

Fig.8. Recommendation using Collaborative Filtering

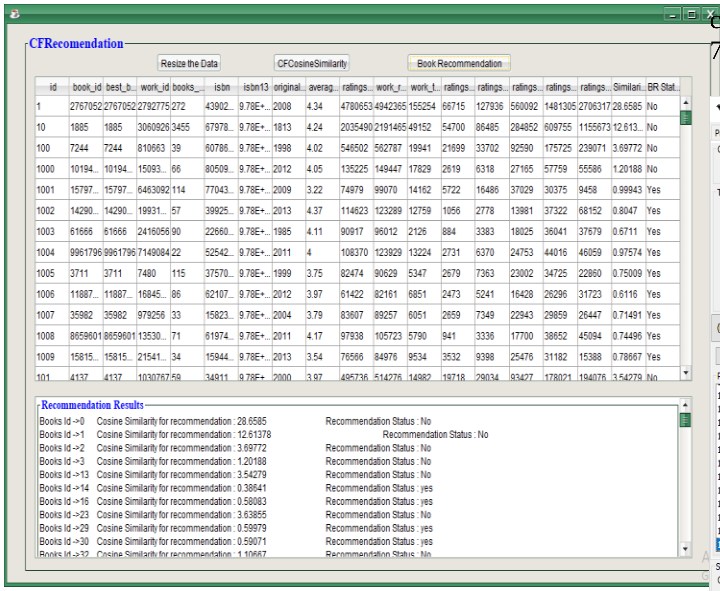

Fig.9. Recommendation status

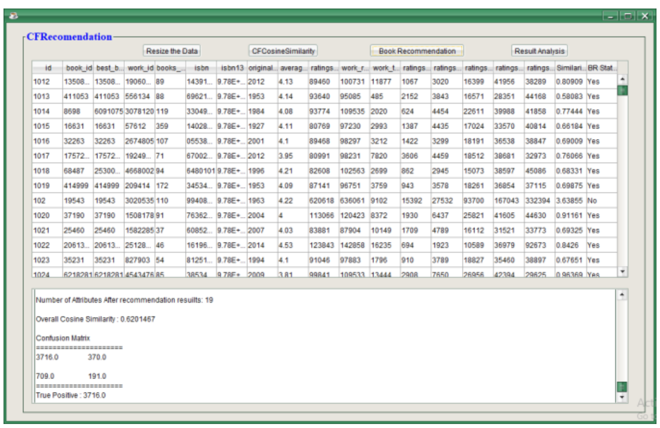

Fig.10. Creation of confusion matrix

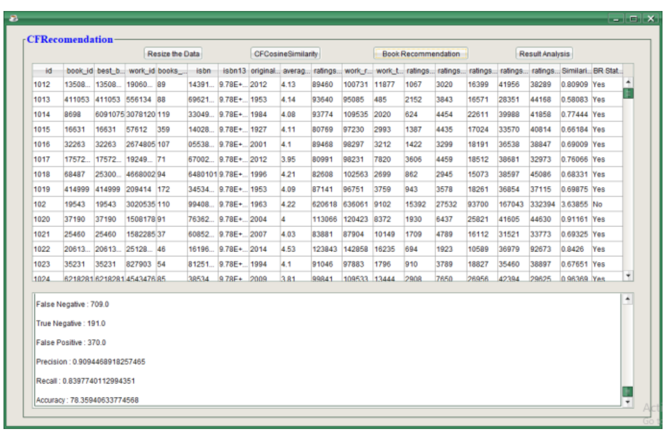

Fig.11. Accuracy prediction

ANN-OLVQ Classification

The Artificial Neural Network OLVQ classification system is simulated through weka libraries. Fig.11 shows the correctly classified instances in terms of accuracy as 79.845%.

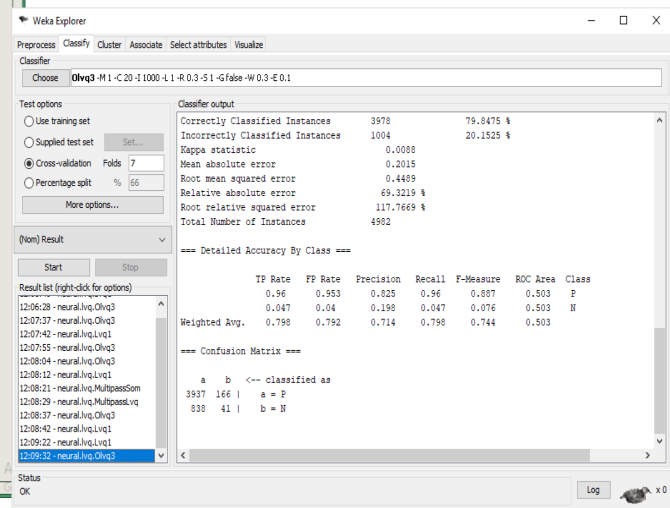

Fig.11 Classification Accuracy of Conventional ANN-OLVQ system

The proposed Artificial Neural Network- LVQ classification system is simulated through weka libraries. Fig.12 shows the correctly classified instances in terms of accuracy as 81.234%.

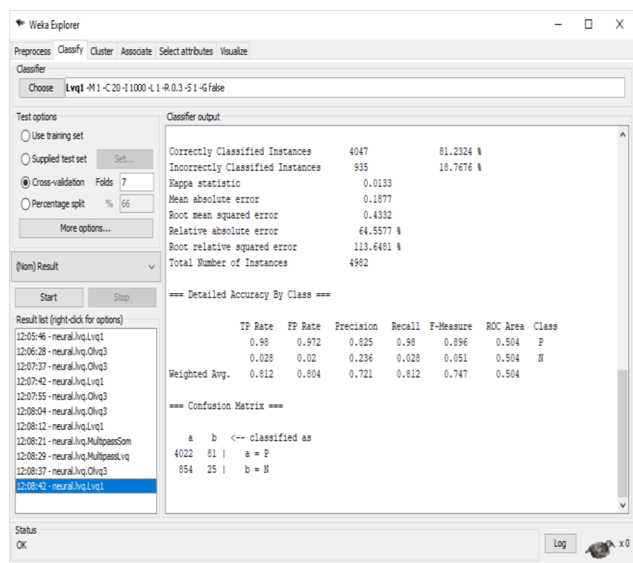

Fig.12. Classification Accuracy of the proposed ANN- LVQ system

Performance Matrices

Various performance indices such accuracy, error rate, precision-recall and execution time of the proposed ANNOLVQ and ANNLVQ recommender systemcompared with traditional CF recommendation system.

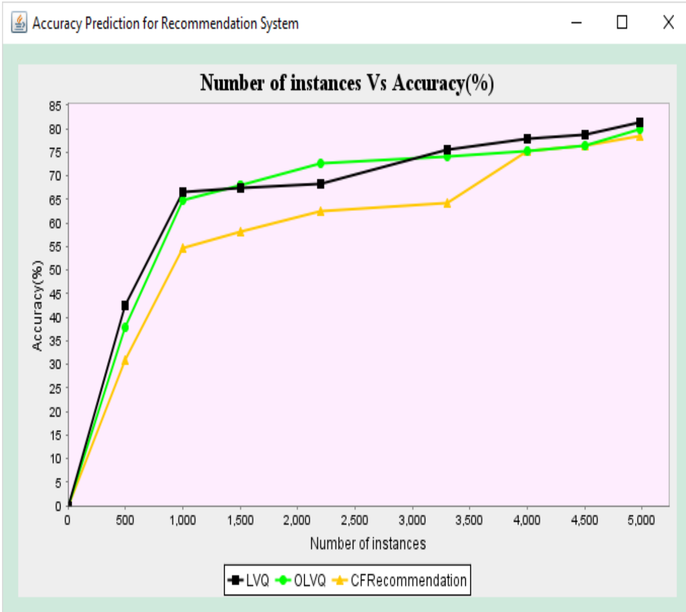

Fig.13. Number of instances vs. Accuracy

Fig.13 shows that, the predicted accuracy among Artificial neural network LVQ, OLVQ and CF Recommendation based on the number of instances. From this it is shown that, ANNLVQ predicts high accuracy compared withANNOLVQ and CF Recommendation systems.

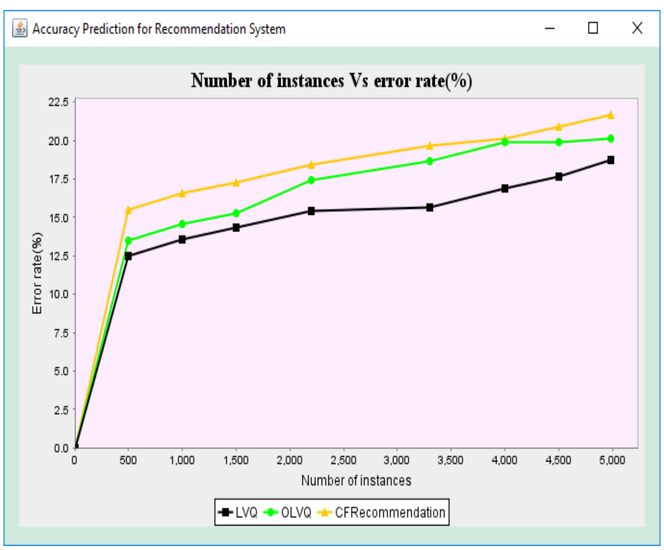

Fig.14. Number of instances vs. Error rate

Fig.14 shows that, the error rate among Artificial neural network LVQ, OLVQ and CF Recommendation based on the number of instances. From this it is shown that, ANNLVQ predicts have low error rate compared CF and ANNOLVQ Recommendation systems.

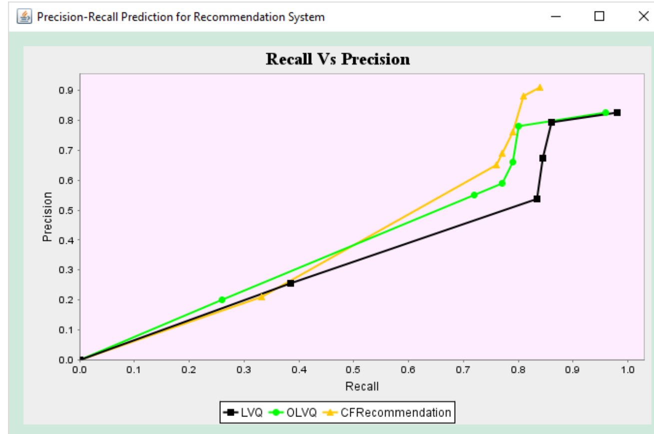

Fig.15. Comparison of Recall vs. Precision

Fig.15 shows the comparison among Artificial neural network LVQ, OLVQ and CF Recommendation based on the precision and recall. From this it is shown that, LVQ predicts have high precision vs recall characteristics compared with OLVQ and CR recommendation systems.

Fig.16 shows the number of instances vs execution time characteristics among Artificial neural network LVQ, OLVQ and CF Recommendation based on the precision and recall. From this it is shown that, LVQ predicts have low execution time compared with CF and OLVQ.

Based on the results of the performance matrices depicted fig.13- fig.16, it is stated that the proposed recommender ANNOLVQ based recommender system effectively classified the class of the book based on the reviewer comments. The classification accuracy of the ANNOLVQ is high while compared with the existing LVQ and CF recommendation systems.

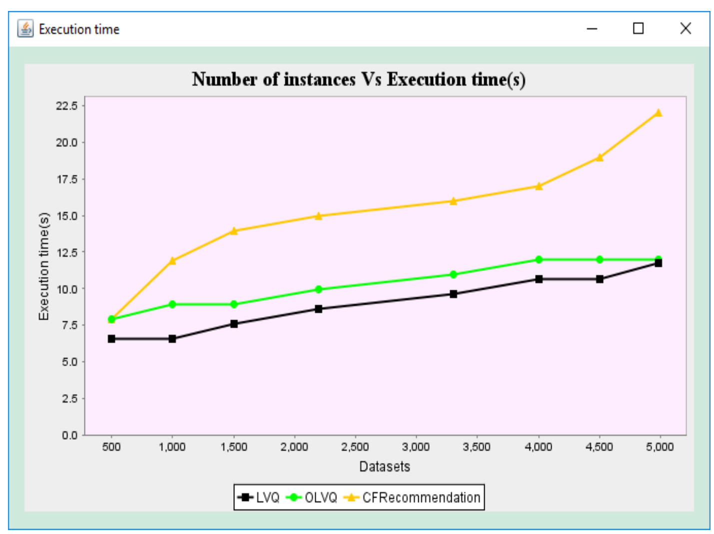

Fig.16. Number of instances vs execution time characteristics

V. CONCLUSION

This paper proposed Artificial Neural Network approach to develop a multi-categorical classification model to predict user ratings. In this proposed research, the problem of predicting the rating as a multi-label classification problem was considered where each rating had treated a label. From the results, it is showed that the ANNLVQ classification approach achieved a high prediction accuracy rate in the book recommender systems. So artificial neural network based recommender system better than traditional recommender system In the future, the performance of the proposed classification system will be analyzed with dynamic datasets in the different application based on the online reviews.

REFERENCES

[1]. Alharthi, H., Inkpen, D., & Szpakowicz, S. A survey of book recommender systems. *Journal of Intelligent Information Systems*, *51*(1), 139-160, (2018).

[2]. Jakomin, M., Curk, T., & Bosnić, Z. Generating inter-dependent data streams for recommender systems. *Simulation Modelling Practice and Theory*, *88*, 1-16,(2018).

[3]. Scholz, M., Dorner, V., Schryen, G., & Benlian, A. A configuration-based recommender system for supporting e-commerce decisions. *European Journal of Operational Research*, *259*(1), 205-215, (2017).

[4]. Lin, K. P., Shen, C. Y., Chang, T. L., & Chang, T. M. A Consumer Review-Driven Recommender Service for Web E-Commerce. In *Service-Oriented Computing and Applications (SOCA), 2017 IEEE 10th International Conference on* (pp. 206-210), 2017.

[5]. Smith, B., & Linden, G. Two decades of recommender systems at Amazon. com. *Ieee internet computing*, *21*(3), 12-18, (2017).

[6]. Zhang, F., Lee, V. E., Jin, R., Garg, S., Choo, K. K. R., Maasberg, M., ... & Cheng, C. Privacy-aware smart city: A case study in collaborative filtering recommender systems. *Journal of Parallel and Distributed Computing*, (2018).

[7]. Kaur, H., Kumar, N., & Batra, S. An efficient multi-party scheme for privacy-preserving collaborative filtering for healthcare recommender system. *Future Generation Computer Systems,* (2018).

[8]. Nilashi, M., Ibrahim, O., & Bagherifard, K. A recommender system based on collaborative filtering using ontology and dimensionality reduction techniques. *Expert Systems with Applications*, *92*, 507-520, (2018).

[9]. Yang, B., Lei, Y., Liu, J., & Li, W. Social collaborative filtering by the trust. *IEEE transactions on pattern analysis and machine intelligence*, *39*(8), 1633-1647, (2017).

[10]. Liu, Y., Wang, S., Khan, M. S., & He, J. A novel deep hybrid recommender system based on auto-encoder with neural collaborative filtering. *Big Data Mining and Analytics*, *1*(3), 211-221, (2018).

[11]. Li, Y., Wang, D., He, H., Jiao, L., & Xue, Y. Mining intrinsic information by matrix factorization-based approaches for collaborative filtering in recommender systems — *Neurocomputing*, *249*, 48-63, (2017).

[12]. Zafari, F., & Moser, I. Modeling socially-influenced conditional preferences over feature values in recommender systems based on factorized collaborative filtering. *Expert Systems with Applications*, *87*, 98-117, (2017).

[13]. Bampis, C. G., Rusu, C., Hajj, H., & Bovik, A. C. Robust Matrix Factorization for Collaborative Filtering in Recommender Systems. In *Asilomar Conf. on Signals, Systems, and Computers*, (2017).

[14]. Mazze, A. (2017). Recommender system using ANN. *Neural Networks & Machine Learning*, *1*(1), 3-3.

[15]. Lee, D. H., & Brusilovsky, P.Improving personalized recommendations using community membership information. *Information Processing & Management*, *53*(5), 1201-1214, (2017).

[16]. Suglia, A., Greco, C., Musto, C., de Gemmis, M., Lops, P., & Semeraro, G. A deep architecture for content-based recommendations exploiting recurrent neural networks. In *Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization* (pp. 202-211). ACM, (2017).

[17]. Devooght, R., & Bersini, H. Long and short-term recommendations with recurrent neural networks. In *Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization* (pp. 13-21). ACM, (2017).

[18]. Yi, B., Shen, X., Zhang, Z., Shu, J., & Liu, H. Expanded autoencoder recommendation framework and its application in movie recommendation. In *Software, Knowledge, Information Management & Applications (SKIMA), 2016 10th International Conference on*(pp. 298-303). IEEE, (2016).

[19]. Veugen, T., & Erkin, Z. Content-based recommendations with approximate integer division. In *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on* (pp. 1802-1806), (2015).

[20]. Shen, X., Yi, B., Zhang, Z., Shu, J., & Liu, H. Automatic Recommendation Technology for Learning Resources with Convolutional Neural Network. In *Educational Technology (ISET), 2016 International Symposium on* (pp. 30-34), (2016).

[21]. Chen, L., & Wang, F. Explaining recommendations based on feature sentiments in product reviews. In *Proceedings of the 22nd International Conference on Intelligent User Interfaces* (pp. 17-28). ACM, (2017).

[22]. Paradarami, T. K., Bastian, N. D., & Wightman, J. L. A hybrid recommender system using artificial neural networks. *Expert Systems with Applications*, *83*, 300-313, (2017).

[23]. Tewari, A. S., & Barman, A. G. Sequencing of items in personalized recommendations using multiple recommendation techniques. *Expert Systems with Applications*, *97*, 70-82, (2018).

[24]. Liu, D. R., Chen, K. Y., Chou, Y. C., & Lee, J. H. Online recommendations based on a dynamic adjustment of recommendation lists. *Knowledge-Based Systems*, (2018).

[25]. de Campos, L. M., Fernández-Luna, J. M., Huete, J. F., & Redondo-Expósito, L. Positive unlabeled learning for building recommender systems in a parliamentary setting. *Information Sciences*, *433*, 221-232, (2018).

[26]. Lee, S. J., Xu, Z., Li, T., & Yang, Y. A novel bagging C4. 5 algorithm based on wrapper feature selection for supporting wise clinical decision making. *Journal of biomedical informatics*, *78*, 144-155, (2018).

[27]. Li, T., Fu, K., Choi, M., Liu, X., & Chen, Y. Toward Robust and Efficient Training of Generative Adversarial Networks with Bayesian Approximation. In *the Approximation Theory and Machine Learning Conference*, (2018).

[28]. Liu, Y., Wang, S., Khan, M. S., & He, J. A novel deep hybrid recommender system based on auto-encoder with neural collaborative filtering. *Big Data Mining and Analytics*, *1*(3), 211-221, (2018).

[29]. Li, Y., Wang, D., He, H., Jiao, L., & Xue, Y. Mining intrinsic information by matrix factorization-based approaches for collaborative filtering in recommender systems — *Neurocomputing*, *249*, 48-63, (2017).

[30]. Zafari, F., & Moser, I. Modeling socially-influenced conditional preferences over feature values in recommender systems based on factorized collaborative filtering. *Expert Systems with Applications*, *87*, 98-117, (2017).

[31]. Bampis, C. G., Rusu, C., Hajj, H., & Bovik, A. C. Robust Matrix Factorization for Collaborative Filtering in Recommender Systems. In *Asilomar Conf. on Signals, Systems, and Computers,* (2017).

[32]. Mazze, A. (2017). Recommender system using ANN. *Neural Networks & Machine Learning*, *1*(1), 3-3.

[33]. Lee, D. H., & Brusilovsky, P.Improving personalized recommendations using community membership information. *Information Processing & Management*, *53*(5), 1201-1214, (2017).

[34]. Suglia, A., Greco, C., Musto, C., de Gemmis, M., Lops, P., & Semeraro, G. A deep architecture for content-based recommendations exploiting recurrent neural networks. In *Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization* (pp. 202-211). ACM, (2017).

[35]. Devooght, R., & Bersini, H. Long and short-term recommendations with recurrent neural networks. In *Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization* (pp. 13-21). ACM, (2017).Yi, B., Shen, X., Zhang, Z., Shu, J., & Liu, H. Expanded autoencoder recommendation framework and its application in movie recommendation. In *Software, Knowledge, Information Management & Applications (SKIMA), 2016 10th International Conference on*(pp. 298-303). IEEE, (2016).

[36]. Veugen, T., & Erkin, Z. Content-based recommendations with approximate integer division. In *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on* (pp. 1802-1806), (2015).

[37]. Shen, X., Yi, B., Zhang, Z., Shu, J., & Liu, H. Automatic Recommendation Technology for Learning Resources with Convolutional Neural Network. In *Educational Technology (ISET), 2016 International Symposium on* (pp. 30-34), (2016).

[38]. Chen, L., & Wang, F. Explaining recommendations based on feature sentiments in product reviews. In *Proceedings of the 22nd International Conference on Intelligent User Interfaces* (pp. 17-28). ACM, (2017).

[39]. Paradarami, T. K., Bastian, N. D., & Wightman, J. L. A hybrid recommender system using artificial neural networks. *Expert Systems with Applications*, *83*, 300-313, (2017).

[40]. Tewari, A. S., & Barman, A. G. Sequencing of items in personalized recommendations using multiple recommendation techniques. *Expert Systems with Applications*, *97*, 70-82, (2018).

[41]. de Campos, L. M., Fernández-Luna, J. M., Huete, J. F., & Redondo-Expósito, L. Positive unlabeled learning for building recommender systems in a parliamentary setting. *Information Sciences*, *433*, 221-232, (2018).

[42]. Lee, S. J., Xu, Z., Li, T., & Yang, Y. A novel bagging C4. 5 algorithm based on wrapper feature selection for supporting wise clinical decision making. *Journal of biomedical informatics*, *78*, 144-155, (2018).

[43]. Li, T., Fu, K., Choi, M., Liu, X., & Chen, Y. Toward Robust and Efficient Training of Generative Adversarial Networks with Bayesian Approximation. In *the Approximation Theory and Machine Learning Conference,* (2018).

[44]. Li, T., Fu, K., Choi, M., Liu, X., & Chen, Y. Toward Robust and Efficient Training of Generative Adversarial Networks with Bayesian Approximation. In *the Approximation Theory and Machine Learning Conference,* (2018).

[45]. Belka, A., Fischer, M., Pohlmann, A., Beer, M., & Höper, D. (LVQ-KNN: Composition-based DNA/RNA binning of short nucleotide sequences utilizing a prototype-based k-nearest neighbor approach. *Virus research*, *258*, 55-63, (2018)

