International Journal of Electrical and Computer Engineering (IJECE)
Vol. 9, No. 5, October 2019, pp. 3813~3821
ISSN: 2088-8708, DOI: 10.11591/ijece.v9i5.pp3813-3821

Recommender System for Surplus Stock Clearance

Vipul Agarwal, Vijayalakshmi A
Department of Computer Science, CHRIST (Deemed to be University), India

Article Info

Article history:

Received Jan 12, 2019
Revised Apr 9, 2019
Accepted Apr 22, 2019

Keywords:

Inventory clearance
Recommender systems
Retail stock management
Stock accumulation
Stock clearance
Surplus stock

ABSTRACT

Accumulation of the stock had been a major concern for retail shop owners. Surplus stock could be minimized if the system could continuously monitor the accumulated stock and recommend those which require clearance. Recommender Systems computes the data, shadowing the manual work and give efficient recommendations to overcome stock accumulation, creating space for new stock for sale to enhance the profit in business. An intelligent recommender system was built that could work with the data and help the shop owners to overcome the issue of surplus stock in a remarkable way. An item-item collaborative filtering technique with Pearson similarity metric was used to draw the similarity between the items and accordingly give recommendations. The results obtained on the dataset highlighted the top-N items using the Pearson similarity and the Cosine similarity. The items having the highest rank had the highest accumulation and required attention to be cleared. The comparison is drawn for the precision and recall obtained by the similarity metrics used. The evaluation of the existing work was done using precision and recall, where the precision obtained was remarkable, while the recall has the scope of increment but in turn, it would reduce the value of precision. Thus, there lies a scope of reducing the stock accumulation with the help of a recommender system and overcome losses to maximize profit.

Corresponding Author:

Vipul Agarwal,
Department of Computer Science,
CHRIST (Deemed to be University),
Hosur Main Road, Bengaluru-560029, Karnataka, India.
Email: vipul.agarwal@mca.christuniversity.in

1. INTRODUCTION

Recommender Systems (RS) have played a vital role to speed up simple day to day activities by providing efficient and useful recommendations be it in the e-commerce domain, downloading songs/ movies, etc. and have never disappointed us. Due to their versatility, they have become an important topic of research and increasingly, started grabbing the user's interest. Stock Management is the basic understanding of the stock mix in a company along with various demands of the stock. There are various factors which influence the stock outflow and inflow that in turn affect the profit and loss in the organization. Although there have been various methods including both the manual work and system calculation, yet the outcome is solely decided by the designated person.

Recommender systems benefit both the user and the provider by saving the time of the user who keeps on searching for an item which they prefer in the application like e-commerce while the providers are benefitted by enhancing their business and popularity, overall increasing their efficiency. The stock management system has been one of the crucial problems which the business must focus on to maximize their profit and minimize loss. With technology coming to picture, the business hubs have migrated from the paperwork to computers in the form of excel sheets but still not making efficient use of the actual data, failing in data analysis. Large business hubs have a huge amount of data in excel sheet but cannot identify the surplus stock in advance or the trend until the yearly, quarterly or the monthly analysis is done, and by the time the results are out, it is too late to react. Moreover, these kinds of work require additional manpower to solve the issue and thus cause additional expense to the business.

Hence, recommender systems for surplus stock clearance are all about managing the stock accumulation, which will study and analyze the data regarding the stock in the inventory and tell about the top-N items being accumulated in the organization. Although excel sheets are present with similar data, yet they have a limit to the amount of data they can process, and they also lack the major advancements/ algorithms which can be obtained through machine learning algorithms than the ones in excel formula computations. Thus, one must know the use of recommender system in this scenario.

In this paper, the use of recommendation system is being proposed for stock clearance, which will compute the top-N items which have accumulated in the stock and needs to be cleared to minimize the losses due to surplus stock. A python script has been written which trains the recommender system with the existing dataset, and then the trained system is fed with the new dataset for increasing the efficiency of the prediction made by the algorithm. The algorithm uses Pearson's similarity as the metric for the item-item collaborative filtering technique to draw the similarity between similar kind of items. Once the data is fed, the corresponding results are obtained as the top-N items which the user must focus to clear the stock to prevent losses and maximize profit.

Along with this, the system is also evaluated with precision and recall, understanding how well the system is predicting and accordingly draw the conclusions for the system built. Such kind of system doesn't exist; thus, it is a novel idea being proposed to ease the work of the stock managers and maximize profit by overcoming the losses due to stock accumulation.

Before creating a recommender system, one must be familiar with the basic knowledge about a recommender system and how it functions so that while replicating the same, the researcher understands what, when and why about the recommender system. The recommendation engine works efficiently if the phases provide a successive output for the next phase. If the output of any of these phases is compromised, then it directly affects the quality of the recommendations. There are three phases of the recommendation process:

a. Information Collection Phase

This phase involves the collection of related or relevant information [1] about the user profiles or models or items for the prediction task which involves user's attributes, behaviors, likes, dislikes, explicit feedbacks, etc.

b. Learning Phase

This phase requires a learning algorithm to filter and exploit the user's features from the feedback gathered in the information collection phase and in turn find out about the relevance in the existing information.

c. Prediction or Recommendation Phase

This phase is also known as the implementation phase which recommends or predicts what kind of items the user may prefer, depending on the information collected in the first phase and analysis involved (by using the algorithms, existing ones or the new ones) in the second phase.

2. RECOMMENDATION FILTERING TECHNIQUES

The recommendation system solely depends on the different filtering techniques used to get efficient recommendations. There are several techniques available and each technique serves different purposes and highlights their potential in various domains depending on the requirement.

2.1. Content-based filtering

In this filtering technique, a match is made between the description of items and description given by users for the item which they are searching. The recommendations solely lie on what the current user seeks depending on its description rather than giving a recommendation based on the other users of the same kind, like that in collaborative technique. It involves our work on the metadata. This process does not require the profile of other users as it does not influence recommendations. If the current users' profile changes, the technique has the capability to adjust its recommendations in negligible time. The main disadvantage of this technique is that it must have in-depth knowledge about the description of all the features in the profile else the performance goes down significantly.

2.2. Collaborative filtering (CF)

This technique gives recommendations based upon the similarities between the user and between the items being searched depending on explicit relevance (ratings, tags, etc.) or implicit relevance (actions involved like reading, downloading, etc.). It depends on the preferences of other users to draw recommendations. A user will get a recommendation of those items that have not been rated before but were rated by other users in the neighborhood i.e. those users who have similar interests and preferences which is calculated by the similarities between their profiles. As per Figure 1, we can see the basic understanding of the CF and content-based filtering. This technique is one of the most widely implemented.

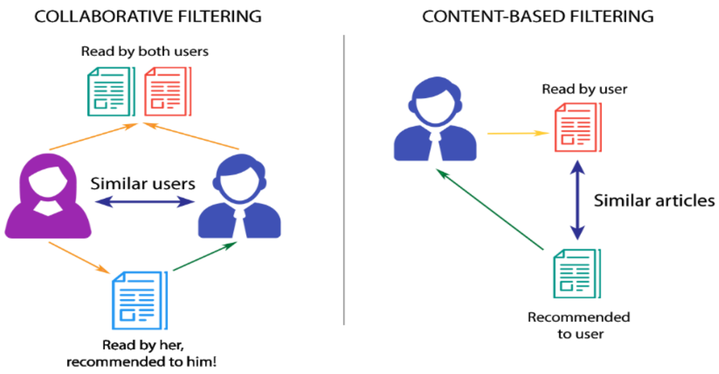

Figure 1. Collaborative filtering and content-based filtering

3. RELATED WORK

Recommender system has been a major topic of research over the past decades. Various algorithms have been deployed extensively in various domains to enhance the performance of the recommender systems. The human-recommender interactions based on visualization framework [2] which combines with visualization technique was proposed. Depending on further analysis of the existing systems and result surveys, the paper draws future research challenges and opportunities in this area. A survey on e-commerce sites [3] using RS was performed to know how they are being benefited by increasing their sales and enhancing their profit. It also describes various sites which have incorporated more than one recommender system. A brief study is made on the inputs required by them from the customers, the recommendation techniques and the interface provided to the users. E-commerce has allowed people to choose from multiple options, breaking the stereotype of mass production.

Web usage clustering has been a popular area of research wherein different techniques are being used to get the maximum output from web usage [4]. The focus lies on overcoming the obstacles, giving the correct result of web usage clustering using recommendation systems. The process of finding, analyzing, pre-processing and efficient clustering of data helps in the building efficient recommender systems. The author proposes an analysis and pre-processing model to come out of the poor quality of web usage data, whereas to ensure the efficiency of the clustering, Particle Swarm Optimization (PSO) approach is used. Commonly known as KDD (Knowledge Discovery in Data mining), the term holds great importance in the extraction of useful and relevant information form a large data set. These work on the basis of clustering related data, patterns, prediction of requirements of users and giving valid information. Thus, a study of their requirement, the patterns followed by them and their interests is essential for web usage clustering. discusses the different image classification algorithms.

The importance of clustering of web sessions [5] is drawn, which is an integral part of the mining technique to group the sessions based on some similarity between them. The paper also gives a new algorithm for Particle Swarm Optimization (PSO). The proposed algorithm has no link with other existing clustering algorithms. The main job of web session clustering is to extract the total usage of the web and the pattern in which the user navigates between web pages and predict the behavior.

An Ultra Large Scale (ULS) software projects [6] are considered to have high complexity, yet the requirements and needs of these projects are not met. Thus, a process is being established to use the data mining and recommender systems to involve the stakeholders of such large projects. Data Mining is considered as a useful technique wherein the data is collected, and they are structured and categorized for easy processing and outcome. Attention is drawn towards the highlights of the recommendations on personalized products [7] and to cross the hurdle of information overload when the amount of data involved is huge.

Over the years, E-commerce has functioned differently and now provides recommendations more likely and efficient, enhancing their business. Recommender systems in e-commerce [8] portrays the use of the emerging system in the e-commerce industry, one of the fastest growing markets in today's world. To compete with the growing virtual market and the physical market, the system is adapted in such a way that tough competition is ensured between the two industries, having the same goals but different approach. Thus, it draws an insight into the techniques to provide recommendations to the customers.

A movie-based recommendation system [9] is a boon for society. The k-mean cluster is used in this case wherein the cluster heads are made of similar kind of items and grouped together. Each observation is broken into k-clusters where each cluster belongs to the cluster having the nearest mean, which satisfies the property of the cluster. Hence, using this, an attempt is made to increase the efficiency of the system.

Such a pool of uses of recommender systems has been discussed along with their advantages and disadvantages in their respective field. The cold-start issue as we know is one of the major drawbacks in any recommender systems. Here we classify cold-start problem into two categories, i.e. cold-user and cold-item. A cold-user is one who is a new user [10] to the system. For instance, when a user creates a new account in any of the e-commerce sites for buying an item, the system is unable to interact with the user to provide a recommendation as there is data inadequacy about the user's history, likes, dislikes and previous purchases in other similar domains. There is an attempt to reduce the sparsity issue also i.e. reducing the sparsity of the sparse matrix. This would fill in the holes in the sparse matrix so that there is no data insufficiency and the cold start issue is overcome.

4. DATASET OVERVIEW

A dataset is built for implementing the proposed solution for overcoming the accumulation of stock. The dataset includes the following set of attributes: user id, item id, purchased, stock and date. The dataset includes the item-id of let's say, detergents of different brands. The data is stored in the form of rows, wherein each row has entries for the above attributes, i.e. item-wise the details are stored, and the user-id indicates information about which of the customer has purchased the corresponding item along with the date and the remaining stock of that item. The dataset has entries for a period of over two years. The attribute "stock", is the main attribute of the dataset. This attribute will store the count of every product which is left in the inventory. Hence the system shall rely on this main attribute to provide an efficient recommendation.

5. PROPOSED METHODOLOGY

There are recommendations systems built for e-commerce, movie, etc. In this research work, based on the literature review, a system is designed that gives recommendations for clearing the surplus stock in the inventory in retail shops, malls, and mega-marts. The system shall recommend top-N items having a maximum accumulation in the inventory. As data is generated at a drastic speed, recommender systems have significantly reduced the manual work and drastically demonstrates the required output by providing efficient recommendations in various domains like e-commerce sites (Amazon, Flipkart), entertainment (Netflix), etc. Figure 2 demonstrates an architecture diagram for the proposed work.

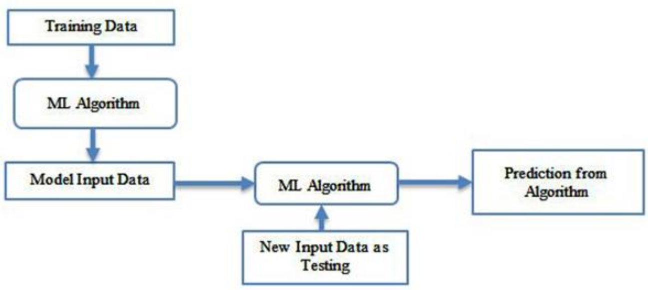

Figure 2. The proposed method for the Item-Item similarity metric

The above architecture begins with the dataset which should be divided into the training data and the testing data, so to go with the usual technique used in machine learning, the first half or quarter of the dataset can be used as the training data and the remaining can be used as the testing data. Based on this, once the model is trained, we provide the model input data and the new input data as the testing data to the trained machine learning algorithm. The results obtained are the predictions of the top-N items having the highest stock accumulation, which needs to be cleared, ranked from the highest priority to the lowest priority to avoid losses.

Another prospect to this may be seen if we consider for a longer period. If these items have poor outflow in a year but gradually improves in the consequent years, the recommender learns this, and although a particular item may have lesser sale compared to another item, the sale gradually picks up, thus the rank of popularity changes for that item despite having more stock as in future, and the sale might increase for that item.

Hence, the stores can come up with their own set of mechanism to overcome stock accumulation as they will be getting a prior recommendation to cope up with the plenty stock available in the inventory. Hence for each category like biscuits, detergents, chocolates, juices, etc., the recommendation engine can provide great results to clear the surplus stock.

It is seen that the target attribute "Stock" will play a vital role in analyzing the existing stock in the inventory. The item-item similarity using Pearson's correlation as it finds the relationship between two variables, i.e., in other words, it indicates the strength of the relationship. We are also trying the same with Cosine similarity where the Cosine is the angle between any two item vectors, indicating that closer the vector, larger will the Cosine and smaller the angle. The similarity will be drawn for items of the same category and thus item-item matrix will be created by the algorithm to process further. This matrix will help to identify the similar kind of items and figure out which is the maximum stock left i.e. showing the least outflow. Hence, an efficient item-similarity metric can be used to create a recommendation engine for clearing the stock from the inventory and overcome losses and maximizing the profit. A system like this does not exist, hence it can be incorporated in the sales domain for retail shops, malls, and mega-marts.

6. IMPLEMENTATION AND RESULTS

In this paper, the item-item collaborative filtering is being used wherein an item look alike matrix is created, which will be further processed, using which recommendation can be given. Hence, with the similar kind of items, we get the top-N recommendation of those items which have accumulated over a period and accordingly different mechanisms can be adopted by the management to take care of their inventory. The algorithm creates a similarity matrix by the pair of items that have stock of items decreasing by the purchases made by the users to ease the process and make the model more reliable. The pseudocode of the script used is given below:

Step 1: Import necessary libraries
Step 2: Read the dataset file
Step 3: Partition the dataset into a 2:1 ratio for training and testing respectively.
Step 4: Train the model with the Training data with "target=purchased" and "similarity_type=Pearson".
Step 5: Train the model with the Training data with "target=purchased" and "similarity_type=Cosine".
Step 6: Make predictions with the Test data and print top-N items.
Step 7: Evaluate precision_recall.

To avoid common mistakes in replicating or creating a recommender system, for this problem in specific or others in general, one must make sure that the training data and testing data are divided in at least 2:1 ratio. i.e. a higher number of data rows for the training dataset and a lesser number of data rows for the testing dataset. This provides an effective training of the recommender system and gives better results irrespective of the size of the dataset used for the system in the future. Also, the dataset must be pre-processed thoroughly to avoid missing values, outliers, or any sort of noise, although noise won't impact the system much still it is advised to remove them from the dataset for smooth functioning.

The script created is a bit time consuming, on average around 2 minutes for the existing dataset, and it may vary due to the amount of data it will process to give a recommendation. A system with better hardware configuration, which most of the people have in today's time, the processing will be much faster than the one done here as it was deliberately worked on a standard system to set an example for the end users that minimal requirements will serve the same purpose as those with the latest technology. On executing the script, the following output is obtained:

Table 1 shows that from the various number of items, the rank points out to the status of items being sold, i.e. rank 1 indicates the item-id 103 has been significantly been accumulated in the inventory, while the item-id 101 with rank 5, has got the least accumulation of the stock in the inventory. Next, we try to execute the script with the Cosine Similarity where Table 2 shows that from the various number of items, the rank points out to the status of items being sold, i.e. rank 1 indicates the item-id 103 has been significantly been accumulated in the inventory, while the item-id 101 with rank 5, has got the least accumulation of the stock in the inventory. However, there is a flip in the rank of item-id 102 and item-id 104. Now let us refer to Figure 3 and get an understanding of the below-obtained tables.

Table 1. The rank associated with items in stock using Pearson's similarity

Item-id	Rank
103	1
104	2
102	3
105	4
101	5

Table 2. The rank associated with items in stock using Cosine similarity

Item-id	Rank
103	1
102	2
104	3
105	4
101	5

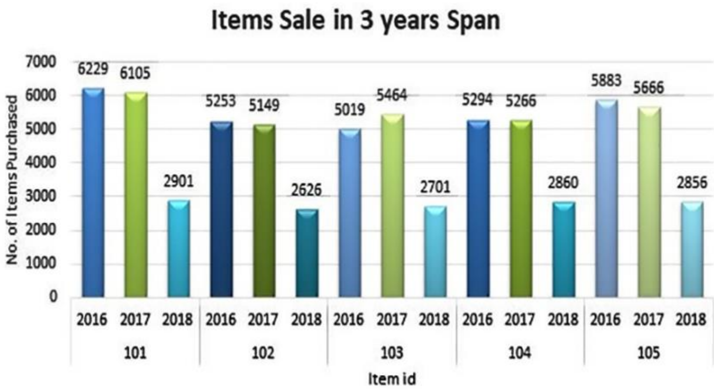

Figure 3. Year wise item sale

Comparing the graph data with the result obtained in Table 1, we can see that although there is a significant sale of the item-id 103 compared with item-id 104 and 102, yet it has rank 1. This is because the recommender system learns, or in other words, we can say that the algorithms learn from the data and thus they can identify that the stock accumulation is increasing for item-id 103 while it is significantly decreasing for item-id 104 and item-id 102.

In Table 2, we see that item-id 103 retains the 1st rank with Cosine similarity as the metric being used, but there is a swap in the rank of item-id 104 and item-id 102. However, the Cosine similarity computes that the item-id 102 needs to be cleared with rank 2, while the item-id 104 holds rank 3.

With the dataset, it is evident that Pearson performs better when compared to the Cosine. The Pearson similarity has taken the consideration of the previous year's same period sales into account and given the ranking accordingly, while the Cosine similarity considers the angles between the items and just focuses on the immediate period and draws the recommendation. Although not much difference is there, the accuracy is what matters for the system.

For a small dataset, the graph can be used but the graph won't be a good learner for predicting the future or trends when compared to the job done by a recommender system. Also, the graph shall serve well with a smaller dataset but in today's scenario, the data is drastically increasing, and the job of the recommender system is to deal with huge amount of data and learn efficiently to provide better recommendation and accurate results so that the people can rely on such systems to overcome losses by stock accumulation.

The overall outcome points towards the item-item similarity recommendation of the collaborative filtering shown below from the dataset which has the maximum number of stocks. The items which have the least outflow from the inventory has been obtained as rank 1 followed by the items with the maximum outflow with rank 5. To summarize it, the items with a higher rank, i.e. rank 1 needs some strategies to be adopted by the retail shops, malls, etc. so that these stocks don't become dead-stock.

Evaluation of results obtained is an important criterion to know how well our system is performing, or in other words to know whether the results obtained are correct or not. The most popular technique used to evaluate the results in the recommender system is precision and recall. Precision(P) is defined as the number of True Positives (TP) over the number of TP plus the number of False Positives (FP), given by the following (1) and (2).

$$P = \frac{TP}{TP+FP} \tag{1}$$

Recall(R) is defined as the number of True Positives (TP) over the number of TP plus the number of False Negatives (FN), given by the following (2):

$$R = \frac{TP}{TP+FN} \quad (2)$$

Thus, we use the precision and recall summary statistics by cut-off as shown in Table 3 for Pearson's similarity metric. Similarly, we use the precision and recall summary statistics by cut-off as shown in Table 4 for Cosine similarity metric.

Table 3. Precision and recall summary using Pearson similarity

Cutoff	Mean_precision	Mean_recall
1	1.0	0.0304
2	1.0	0.0608
3	1.0	0.0912
4	1.0	0.1216
5	1.0	0.1521
6	1.0	0.1522
7	1.0	0.1524
8	1.0	0.1524
9	1.0	0.1524
10	1.0	0.1525

Table 4. Precision and recall summary using Cosine similarity

Cutoff	Mean_precision	Mean_recall
1	0.8	0.0046
2	0.8	0.0068
3	0.8	0.0090
4	0.8	0.0164
5	0.8	0.0322
6	0.8	0.0387
7	0.8	0.0411
8	0.8	0.0411
9	0.8	0.0411
10	0.8	0.0411

Here the cut-off rank ranges from 1-10 and for each rank, the mean_precison and mean_recall are given. For the top-N items where we have taken for 5 items, we have the mean_precison and mean_recall given in the table. To understand cut-off rank, we can say that precision or recall at a given cut-off rank, keeping into account the results obtained are top-N which are returned by the system, thus it is called as precision at n or P@n. We have obtained the cut-off till 10, which means that for the top 10 ranks, the corresponding precision and recall will be obtained.

From Table 3, it is seen that for the Pearson similarity metric, the mean_precision is 1.0 throughout, while the mean_recall increases significantly. As explained, a mean_precision of 1 is a very good result. When compared to Table 4 for cutoff by Cosine similarity, the mean_precision is 0.8 which is lower than the mean_precision obtained through the Pearson similarity metric. Similarly, we see the recall has decreased for the Cosine metric when compared with the Pearson metric.

Higher the value of precision, it gives most of the predicted labels as correct, while low recall indicates that most of its predicted labels are incorrect. Systems with high recall but low precision returns many results, but most of its predicted labels are incorrect when compared to the training labels. A system with high precision but the low recall is just the opposite which will return results, where most of its predicted labels are correct when compared of the training labels.

In an information retrieval system, a precision score of perfect 1.0 indicates that every result which was obtained by the search was relevant but doesn't say anything about whether all the relevant information was retrieved or not. On the contrary, a recall score of perfect 1.0 indicates that all the results obtained by the search was relevant but does not say how many irrelevant results were also obtained.

To conclude, Pearson's similarity gave better results when compared with the Cosine similarity by having a better precision and recall value over the one obtained by the Cosine similarity.

Hence, the system gives a perfect score of precision i.e. 1.0 and indicate all the relevant recommendations are obtained. The recall value can be increased but it would lead to a significant decrease in the precision value. It is a challenge for the researchers to have a high value of both the precision and recall at the same time, but it's still not achieved and thus it leaves behind a scope for improvement in every domain.

7. CONCLUSION

Recommendation system has proved to be of great support in various domains to filter out the content and provide useful recommendations making the work of the users easier. In this paper, the design of a recommender system for the surplus stock is described which will help the retailers in identifying the item that can lead to the accumulation of stocks. This method implements an item-item collaborative filtering technique, with the Pearson similarity metric and the Cosine similarity metric which will be used for identifying the top-N items as mentioned in the results of the similar category which have the highest stock accumulation stating rank 1 in the inventory over a period and give an efficient recommendation about the item-id to clear the existing stock. The precision and recall obtained by the Pearson and the Cosine similarity metric also indicate the performance of Pearson similarity metric trumps over that of the Cosine similarity metric. The precision and recall also indicate the performance of the system and there is always a scope to enhance the performance which is a challenge for the researchers. Experiences of this research, problems, and come back for the problems has been shared along with the results so that the researchers don't face any trouble in duplicating the same or creating a new system of a similar kind. This, in turn, broadens the scope of researchers to learn more and overcome the existing problems, identifying domains where recommender systems have not been implemented. Along with that, a comparison can be made with other techniques and the best one can be discussed and implemented as a future of the existing work done. Thus, one can create a recommendation system which will be more efficient in handling the issues and kill the manual work pressure than those existing in the present system.

ACKNOWLEDGEMENTS

We would like to thank the Almighty for throwing light throughout our tough phases which we came across during our work, without his blessings, it wouldn't have been possible to achieve it. We extend warm gratitude to our supporters and motivators for having confidence in us and providing unending support in every possible way to overcome all the hurdles in this work and stand out. Lastly, we would like to thank our parents for having trust in us and providing all the encouragement and motivation required to accomplish the objective.

REFERENCES

[1] F. Isinkaye, Y. Folajimi, and B. Ojokoh, "Recommendation systems: Principles, methods and evaluation," *Egyptian Informatics Journal*, vol. 16, no. 3, pp. 261-273, 2015.

[2] C. He, D. Parra, and K. Verbert, "Interactive recommender systems: A survey of the state of the art and future research challenges and opportunities," *Expert Systems with Applications*, vol. 56, pp. 9-27, 2016.

[3] J. B. Schafer, J. Konstan, and J. Riedi, "Recommender systems in e-commerce," *Proceedings of the 1st ACM conference on Electronic commerce - EC 99*, pp. 158-166, 1999.

[4] S. Alam, "Intelligent web usage clustering based recommender system," *Proceedings of the fifth ACM conference on Recommender systems - RecSys 11*, pp. 367-370, 2011.

[5] S. Alam, G. Dobbie, and P. Riddle, "Particle Swarm Optimization Based Clustering of Web Usage Data," *2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, pp. 451-454, 2008.

[6] J. Cleland-Huang and B. Mobasher, "Using data mining and recommender systems to scale up the requirements process," *Proceedings of the 2nd international workshop on Ultra-large-scale software-intensive systems - ULSSIS 08*, pp. 3-6, 2008.

[7] Y. H. Cho, J. K. Kim, and S. H. Kim, "A personalized recommender system based on web usage mining and decision tree induction," *Expert Systems with Applications*, vol. 23, no. 3, pp. 329-342, 2002.

[8] S. Sivapalan, A. Sadeghian, H. Rahnama, and A. M. Madni, "Recommender systems in e-commerce," *2014 World Automation Congress (WAC)*, pp. 179-184, 2014.

[9] P. Bargah and N. Mishra, "Solving Sparsity Problem in Movie Based recommendation system," *Advances in Image and Video Processing*, vol. 4, no. 3, pp. 25-31, 2016.

[10] K.R. Bindu, Rhama Lalgudi Visweswaran, P.C. Sachin, Kundavai Devi Solai and Soundarya Gunasekaran, K. R. Bindu, R. L. Visweswaran, P. C. Sachin, K. D. Solai, and S. Gunasekaran, "Reducing the Cold-User and Cold-Item Problem in Recommender System by Reducing the Sparsity of the Sparse Matrix and Addressing the Diversity-Accuracy Problem," *Advances in Intelligent Systems and Computing Proceedings of International Conference on Communication and Networks*, pp. 561-570, 2017.

BIOGRAPHIES OF AUTHORS

Vipul Agarwal is a postgraduate student in Masters of Computer Application at CHRIST (Deemed to be University), Bangalore. My areas of interests are Data Analytics and Machine Learning. Apart from these, my major work and projects are in recommender systems and pattern recognition.

Vijayalakshmi A, works as an Assistant professor at CHRIST (Deemed to be University), Bangalore. She has completed her Ph.D. in face recognition. Her fields of interest include pattern recognition, Machine learning, and Artificial Intelligence. Currently, she is doing a project on IoT and machine learning.