JIMMA UNIVERSITY

INSTITUTE OF TECHNOLOGY

FACULTY OF COMPUTING AND INFORMATICS

Book Recommendation System for New User Using Collaborative Filtering With Demographic Data

By

Emebet Kassa Mussa

A THESIS SUBMITTED TO FACULTY OF COMPUTING AND INFORMATICS, JIMMA UNIVERSITY, IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTERS OF SCIENCE IN COMPUTER NETWORKING

Jimma, Ethiopia
December; 2021

JIMMA UNIVERSITY

INSTITUTE OF TECHNOLOGY

FACULTY OF COMPUTING AND INFORMATICS

Book Recommendation System for New User Using Collaborative Filtering With Demographic Data

By

Emebet Kassa Mussa

Co-advisor:

Teferi Kibebe (MSc)

This is to certify that the thesis prepared by *Emebet Kassa*, titled: ***Book Recommendation System for New User Using Collaborative Filtering with Demographic Data*** and submitted in partial fulfillment of the requirements for the Degree of Master of Science in computer networking complies with the regulations of the University and meets the accepted standards to originality and quality.

Approved by the Examining Committee:

Name **Signature**

1. Mr Teferi Kibebe, Co-Advisor ________
2. Dr. Melkamu Deressa, External examiner ________
3. Ms Kasech Tsegaye , Internal examiner _______________
4. Dr. Faiz Akram _________________

Dedication

This thesis work is dedicated to the Almighty God for Supremacy and the redeemer of my soul.

Acknowledgements

I would like to extend my sincere gratitude to everyone who contributed to this research. First and foremost, I would like to express my appreciation to my co-advisor, Mr Teferi Kebebe, for providing me with good guidance and feedback.

My deepest appreciation would go to my family, especially my husband, for their endless love and unconditional support. Over the years, they did more for me than I can ever pay back. Without their care and encouragement, this thesis would not have been completed.

Additionally, I would like to thank my friends and my staff members for their generous support from the beginning to the end of my thesis works.

Finally, thanks to all those who provided immense support, valuable advice, and inspired me through the journey of my MSc.

Table of Contents

List of Figure

List of table

Acronyms

CBF: Content-Based Filtering

CF: Collaborative Filtering

DF: Demographic Filtering

HF: hybrid filtering

RS: Recommender System

UB: Utility-based filtering

Abstract

Finding relevant information on the Internet has become a major issue today due to information overload. Recommender systems are the best solution to this problem. Recommender system is algorithms aimed at suggesting items of interest to users.

There are many techniques proposed in recommender systems. Collaborative filtering is a common method widely used in recommender systems. However, collaborative filtering techniques still have some problems: cold start. In this study, we propose a book recommender system that uses collaborative filtering with demographic data. We applied the user's age, gender, and occupation to find similarities between users. We cluster users by using K-means clustering. Then the recommender system suggests books that were previously interested by users in the group to new users. Extensive experiments are conducted on user ratings and a dataset of books that include users to evaluate the effectiveness of the proposed model. The performance of the proposed model was evaluated using the precision, recall, and F1 Score metrics that support the effectiveness of the proposed model. The proposed model performance is done by two ways of an experiment. The performance of the proposed model performs around 68.05% of Precision, 42.46% of Recall and 52.1% of the average of F1_score for the experiment based on individual user similarity in the system. And also performs around 93.75% of precision, 40.25% of recall and 56.31% F1-score for the similarity of users based on the similarity of users within the same cluster which is better than the first experiment.

CHAPTER ONE

1. Introduction

Nowadays, there is a huge flow of information on the Web that is growing exponentially, providing users/customers with various resources related to services such as products, hotels, and restaurants. Despite these benefits, the vast flow of information presents challenges for users in processing and selecting a large number of available options. This causes information overload problems [1] and makes the decision-making process difficult. In this case, it is important to filter the information to a limited amount based on current user/customer preferences so that the user can make the right decision. [2] Such a filtering process is usually done by Recommendation System (RS). RS was developed to solve the problem of information overload by providing personalized suggestions for services/items according to specific customer preferences.

RS appeared a decade ago and this field is of great importance in science, business and industry. It is widely used in various fields such as shopping (Amazon), music (Pandora), movies (Netflix), travel (Trip Advisor), restaurants (regsvr), people (Facebook), articles (TED) and so on. Recent advances in e-commerce websites have shown that RS has significant advantages in helping users/customers find relevant items that suit their needs and perhaps their tastes.

Recommender systems use a variety of techniques, including collaborative filtering (CF), content-based filtering (CBF), and hybrid (HF) technology. Content-based filtering (CBF) recommender systems use functions extracted from real items to suggest new items to users, while CF is based on the user's previous preferences compared to other user reviews and or ratings. As the name implies, hybrid recommenders use a combination of content-based and collaborative techniques [40].

Of these methods, collaborative filtering is one of the most successful and most widely used in recommender systems [4]. The assumption of this algorithm is to use shared experience or similar interests. The key process is to find a user who is similar to your target user, or a product that is similar to your predicted product. However, the CF algorithm also has some problems, such as a cold start problem [5]. Cold start problems are a problem where the system cannot recommend items to users. All recommender systems require user profiles to be created with user preferences and interests. User profiles are created with the activities and behaviours of the users on the system. Based on the user's history and activity, the system makes decisions and recommends items. The issue occurs when a new user or item enters the system for a user or item that the system does not have enough information to make a decision. For example, if a new user has not expressed his interest in some items and has not yet visited/viewed some items, then it would be hard for the system to build a model on that basis.

This can lead to inaccuracies in assessing similarities between users. Many approaches have been considered to solve existing problems. Embarak [4] have proposed two types of recommendations, namely node recommendation and batch recommendation. And then compared the proposed method with three other alternatives, including the naive filter bot method, the media scout stereotype method, and the triad aspect method to solve the cold start problem. Basiri et al. [5] proposed a new hybrid approach focused on improving the performance of cold start problems. This technique can provide an inexpensive and suitable recommendation. With the improvement of the era, the person`s conduct and private records may be tracked and recorded on social networking websites or online buying websites. This kind of method makes it less difficult and it's far very beneficial for analyzing person preferences. The motivation for this work is to recommend a book for any reader even for the users who have no past rating history in the Recommendation system. Book recommendations are the domain of recommender systems that motivate us to do this thesis. The work is used to suggest books to readers according to their interest in the book from the available source. The books published in our world through the Internet are huge and it is not convenient for the readers to access the book specifically needed by them because of the availability of several books. So, rather than searching for all books, it is good to be recommended by the system only the interested and related books for readers depending on the user’s profile history which is possible with the book Recommender system. This system will provide only the relevant information for the readers of the book depending on their interest. This book recommendation system is used to help the users with their interest in the book and to provide the updated book for the users based on their interest.

Many works have been done previously focusing on other similar areas like recommending popular books, keyword-based. Still, there is a problem with providing the relevant information for new readers or new users who have no history of data in the recommendation system. Our motivation for this study is to recommend the relevant information for new users which have no history in the system. The recommendation system needs the history data of both the item to be recommended and the user's pattern of interest to make a personalized recommendation system [2]. Our proposed system used the Collaborative Filtering approach with demographic data for book recommender to predict the recommended book for readers.

1.1 Statement of the problem

The availability of books from the World Wide Web is not sufficient enough for book readers to access books based on their preferences; due to the availability of an enormous amount of books with different categories of a book on the web. Moreover, users use each search query to access information about their interests every time from different sources of information. However, it is difficult to get a more relevant book out of all available books. The retrieved book using the traditional information retrieval mechanism will provide all related books with the reader's query without taking into account the user's interest. For example, some user's reader only a specific field of a book that is more related to their profession. Whereas, other readers want to read more general fields (i.e., entertainment). Therefore, a recommender scheme in general and a book recommendation system, in particular, plays a crucial role to fill this gap.

Most book recommendation systems used the user's history and content of the book itself; to suggest and recommend a relevant book for particular readers based on the interests he/she has [6]. This user history (user data) is divided into two, explicit data which is collected by the reader's direct activities (i.e. when users give feedbacks directly to the book while she/ he is reading) and implicit data which the system collects by following the reader's activities (i.e. using the reader's link navigation behaviour including the time spent to read a book) in the system. These user histories (user data) are not sufficient enough to suggest and recommend a relevant book to the readers in the scenario where a new user (user's who haven't history) joins the system [6].

The lack of available user history in the situation where new users/ users join the system limits the effectiveness of the book recommender system to suggest a relevant book to this kind of user. Due to this insufficient user history, there is an occurrence of a new user cold-start problem. The new user cold-start problem is a common problem in the recommendation system because of the lack of history of new users in the system [7]. Furthermore, users get irrelevant books that are not related to their preferences.

1.2 Research question

In this thesis, we investigate and examine the following research questions.

1. How to model user demographic information with collaborative filtering?
2. How to figure out this impact on books recommender scheme towards performance accuracy metrics such as precision, recall and F1-score

1.3 Objectives

1.3.1 General Objectives

The general objective of this study is to propose a book Recommender system to solve the problem with new user cold-start problem which will provide a more related book for readers by using a collaborative filtering approach with demographic data.

1.3.2 Specific Objectives

The following specific objectives are developed from the general objective**.**

- To solve the scalability problem using a clustering algorithm
- To cluster users based on their demographic information
- To evaluate the effectiveness of the recommended books.
- To recommend popular books

1.4 Methodology

The methods or techniques to be applied for this study to achieve its objective is done by reviewing different related papers to our study to get what has been done before and to identify the research gaps to be performed by this proposed system. The other methodology we will use in our study is the method of modelling and implementing the architecture. Data collection and data processing is another method we consider in our work. Also, the evaluation method used to evaluate the performance of a system and compare the results to existing systems is another major issue to be implemented.

1.4.1 Literature Review

There are many works done which are more related to our work. So, different works done before were reviewed with their difference in algorithm design and architecture. We reviewed research articles that used some similar algorithms and their approaches for identifying and designing their model.

1.4.2 System Architecture

The system architecture is the core of the work we implemented. The modelling of the system is done by considering the objectives of this study with the logical order of components of this work. The techniques used to implement an appropriate algorithm for this work is designed and discussed for describing our works clearly.

1.4.3 Implementation Tools

The tools used in our work consider all of the processes we use in the study. So, the tools we used for data preparation, for implementing the proposed model algorithm and designing the model. To design our model we used Edraw software and python anaconda programming to process and provide the recommended books based on a reader's profile.

1.4.4 Data collection

The dataset used to evaluate the proposed system is collected from the BX_crossing dataset which is available on the Internet [8]. The collected data from the website consists of information of book and its metadata, user demographic dataset and rating dataset which is the interaction between users and books.

1.4.5 Evaluation methods

The collection of results from the system, observing the result, analyzing them to assess the system performance is necessary. Analyzing the proposed work with performance metrics is used to identify the contribution achieved in the study. It is used to get the performance of the proposed system and the performance of the approaches used. The system performance is evaluated by using the accuracy metrics precision, recall and F1-score on the items retrieved against the user's interest. The collection of results from the system, statistically analyzing the results obtained is done by using tables and charts.

1.5 Scope

The new user's data used for the proposed system is the only demographic information of users when the registration is done for the system. And the system is not functional for the users not registered to the system.

1.6 Significance of the study

Access to useful and relevant information is a pervasive problem today due to information overload. Recommender systems are a special class of personalized systems that rely on information to predict user interest in available products and services. Search behaviour and previously evaluated items or item features [9]. In general, these are intelligent applications created primarily to help users find and make decisions about interacting with large spaces of information through personalized recommendations. RS is used in educational settings, electronics stores, travel tours, restaurants and hospitals and generally helps in the decision-making process to provide specific users with predictions about the right items [10]. Therefore, improving the quality of recommendations can be important for both information service providers and users. Hence, the book RS that was developed is expected to benefit the following bodies:

- Book users especially new users by aiding them in finding relevant information

- Overcome the problem of information overload by giving users access to interesting, novel and relevant books based on their profile.
- The book provider itself since the concern of provider is how best to provide their customers with adequate means of keeping right with the literature of their interest.
- It can also serve as an input for researchers who want to study in this area.

1.7 Document organization

The proposed work is briefly described below**.** In chapter two, we will explain the literature review, chapter three presents' related works, Chapter four presents the proposed system, Chapter five presents Experimentation Result, Discussion and Evaluation work and Finally, Chapter 6 provides conclusions, suggestions and future work.

CHAPTER TWO

2. LITERATURE REVIEW

This chapter provides an overview of existing recommender systems and recommendation algorithms. The purpose of this chapter is to provide a background related to recommender systems, their methods, and techniques.

2.1 Recommender System

Access to useful and relevant information is a pervasive problem today due to information overload. The recommender system is a special class of personalization aimed at predicting user interest in available products and services by relying on the behaviour of information retrieval and the characteristics of previously selected items or item features [9]. In general, these are intelligent applications designed primarily to help users find and make decisions about interacting with large spaces of information through personalized recommendations.

2.2 Recommender system approach

RS uses a set of ratings explicitly created by the user or implicitly derived from the system [28]. Therefore, we use two types of entities, user and items (two-dimensional), to estimate the score function R.

$$R\text{: } User \times Item = Rating$$

Rating for the (user, item) pairs that have not been rated yet by the users. Here *Rating* is an ordered set (e.g., non-negative integers or real numbers within a certain range), and *User* and *Item* are the domains of users and items respectively. Once the function *R* is estimated for the whole *User* × *Item* space, a recommender system can recommend the highest-rated item (or *k* highest-rated items) for each user. The wide recommender systems approach, fall into the following classes: Content-Based (CBF), Collaborative Filtering (CF), demographic-based RS (DF), Knowledge-based RS, utility-based and hybrid (HF).

2.2.1 Content-based Methods

The content-based recommendation method looks up content information about an item and the user and uses some characteristics to identify the user and the item. To recommend new items to users, content-based filtering matches' features to items that users know they are interested in [10]. For example, to recommend a movie to a user, a content-based recommender system in a movie recommendation application is a movie that the user has previously appreciated (a particular actor, director, genre, subject, etc.). Second, only movies that are very similar to the user's taste are recommended.

One of the benefits of content-based filtering is the ability to recommend items that no one has accessed or purchased. This can be used for newly added items to the system. However, the disadvantage is that important relationships can be overlooked because the item is combined with artificial scores rather than users. Most often, the association of people is between the items higher in relevance than the score created by the system developer [11].

The content-based recommendation approach has its roots in information retrieval [12] and information filtering research. Due to the importance and early progress of the information retrieval and filtering community, and the importance of multiple text-based applications, many content-based systems today are recommending items that contain textual information such as Documents, websites (URLs), and Usenet Book massaging. Improvements to the traditional approach to information retrieval come from the use of user profiles that contain information about user profiles, preferences, and needs.

Porter-stemmed Term Frequency/Inverse Document Frequency (TF/IDF) is one of the most common and well-known CBF techniques. This method uses the full text of the document to generate recommendations. In TF / IDF, stemmed frequency is counted and compared to the entire corpus. Recommendations are then made by collating the items with important keywords. As mentioned earlier, the basics of information retrieval form the core of the CBF approach. [13].

2.2.2 Collaborative Method

In contrast to content-based recommender systems, collaborative recommender systems (or collaborative filter systems) collect item ratings from a large number of users and create recommendations based on other users' patterns of interest. The collaborative filtering approach is based on the assumption that users are usually interested in items that other users with similar interests like [14]. Similarities between user interests are calculated using various methods such as Pearson's correlation coefficient. The system collects ratings for each item from different users, either explicitly or through browsing behaviour, and calculates the similarity between user ratings. Ratings can be explicit on a numerical scale or implicit, such as purchases, clicks, and mouse movements. Users are then grouped based on the correlation between them, and future items are recommended to users based on the recommendations of other users in the group [15]. Consider the groups of users U1 to Un and items *I1* to *Im* shown in the table. Table 2.1 Rating given by the users on different items [49]

	I1	*I2*	*I3*	...	*Im*
U1	1	4	4		4
U2	1	3	4		3
U3	2	4	3		5
U4	2	4	3		
U5	2	4			4
...					
Un	3	4	1		4

For example, if the similarity score between users U1 and U5 is high, then users U1 and U5 can be grouped and new items are recommended to each user based on the interests of other users. Here, item I3 was notified to the user U5, as a new item based on the high rating given by the other user in the group U1. Similarly, item *Im* was recommended to user U4 based on the rating of other user U3. The collaborative systems can be used to filter all types of items, including multimedia items. CF typically suffers from data sparsity and the cold start problem, e.g. many users have purchased very different products, and it can be hard to create groups of users with similar preference. Another case is when no one has purchased a particular product, and it gets visited by a user. In this case, it is difficult to find users with similar preferences, because there simply are none. The advantage of the collaborative approach is that the recommendations are reliable when large amounts of data are available. It reflects the actual user behaviour [16]. CF approaches often fall into two categories: *memory-based* and *model-based.*

i. Memory-based CF

In the memory-based CF approach, RS provide recommendations using all rated items held in the memory [17]. In this way, similarities between users are identified by stored ratings and the required predictions are calculated as needed.

There are two types of memory-based approaches in collaborative filtering. One is user-based filtering and the other is item-based filtering. In a *user-based* filtering approach, recommendation items are forecasts based on finding recommender system users with similar item favorites as the active user.

The methodology can be explained in three steps: [17]

1. The recommender system creates a set of similar users for the active user "u" using the similarity measure; the selected K users are the closest (similar) K users to the active user "u".

2. When k close neighbors of active user "u" is found, the prediction of item "i" is made using one of the following aggregation approaches, mean, weighted sum, or weighted aggregate.

3. To get the top n recommendations, select n items from similar items that are close to the active user. User-based collaborative filtering has scalability issues.

Item-based collaborative filtering: As the number of users grows, user-based KNNs suffer from scalability issues. To overcome this disadvantage, the new approach called item-item Knn was introduced by Sarwar et al. [17]. The item-based approach examines a series of items rated by the target user, calculates the similarity to the target item I, and then selects the k most similar items $i1$, $i2$… ik.

At the same time, the similarity of their expressions $ti1$, $ti2$… tik is calculated. The most similar items were previously discovered and the predictions are calculated from the weighted average of the target user ratings for these similar items. Similarity calculations and prediction generation are two key factors that make item-based recommendations more powerful. For similarity computations, different types of similarity are used to calculate similarity, and weighted sums and regressions are used to calculate predictions.

ii. Model-based

On the contrary to the memory-based approach, the model-based approach uses historical data to allow the model to predict the rating of items that new users have not yet rated. Therefore, similarities between users are identified by design models such as clustering models and statistical models.

The major drawback of memory-based filtering is the need to load large amounts of inline memory. The problem becomes more serious when the rating matrix grows in situations where so many people use the system. It consumes a lot of computing resources and reduces system performance. Therefore, the system cannot immediately respond to the user's request. The model-based approach aims to solve these problems. Model-based CF has four common approaches: Markov decision process (MDP, latent models, classification, matrix factorization, and clustering. In this study, we used the clustering approach to cluster user demography by using the K-means clustering algorithm. The detailed collaborative filtering recommender systems are depicted in figure 2.1.

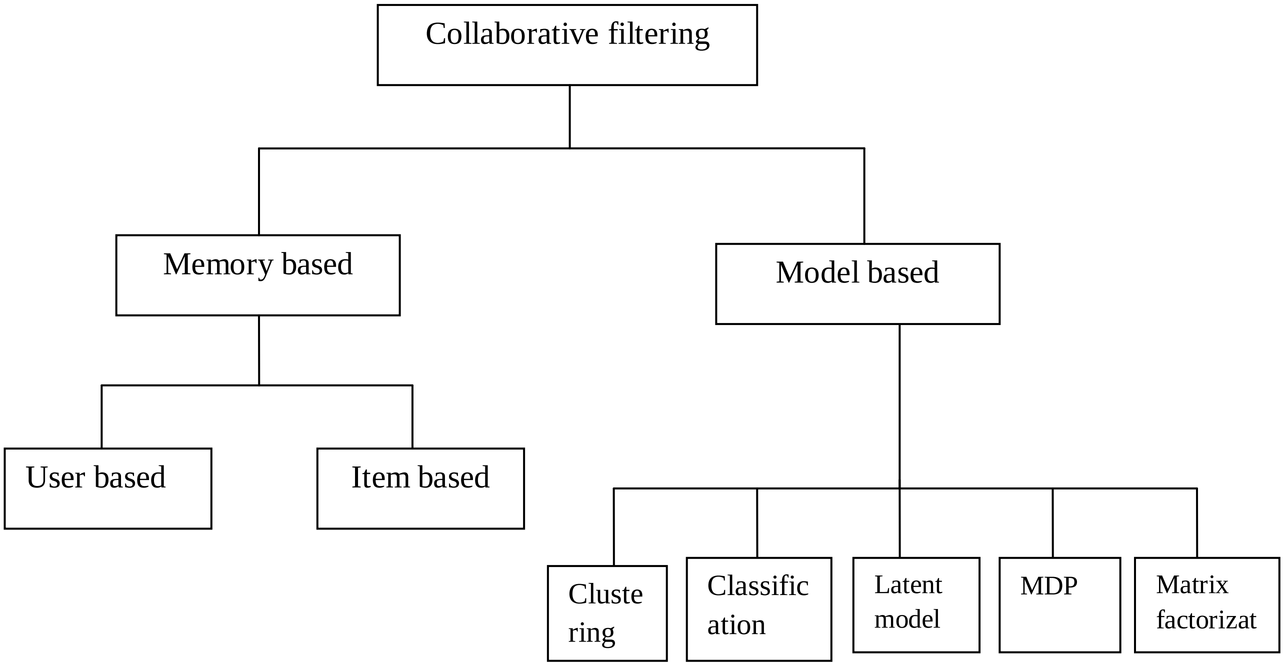


Fig 2. 1 Types of collaborative filtering

2.2.3 Demographic-Based RS

Collaborative filtering techniques can be improved by demographic filtering, as various quantitative research studies have shown [18]. Demographic Recommender System can make recommendations by classifying users based on demographic attributes. The demographic recommender system is particularly useful when the amount of item information is limited. It aims to address and solve scalability and cold start issues. The system uses user attributes as demographic data to get recommendations (that is, item recommendations based on age, sex, occupation, language etc.) [19]. the main advantage of demographic filtering is that you can get results quickly and easily with a few observations. These approaches have also not received the essential user ratings that are mandatory in content-based and collaborative filtering technology.

2.2.4 Utility-Based Recommendation Systems

Utility-based RS provides recommendations based on the generation of a utility model for each item in the user. The system creates a utility function for multi-attribute users and explicitly recommends the item with the highest utility value based on the calculated user utility of each item [20]. Utility-based RS is useful because you can incorporate non-product attributes into utility functions such as Product availability and supplier reliability. They generate utility calculations that allow you to see both inventory and properties of an item in real-time. This allows users to visualize their status. Utility-based systems do not support the long-term generalization of users. Instead, evaluate recommendations based on the user's current needs and available options. The drawbacks of utility-based systems occur when the product is not sufficiently descriptive. It does not contain enough utility features. This can hide recommendations from users, even if they match the preferences of a particular user [21].

2.2.5 Knowledge-Based Recommendation Systems

These types of recommender systems use explicit knowledge of products and users, Knowledge base criteria for generating recommendations [22]. A knowledge-based Recommender system does not initially require huge amounts of data because the recommendations are independent of user ratings [21]. We evaluate products that meet the needs of users and recommend products that meet the tastes of users. A knowledge-based recommender system is useful for several purposes. For example, you can avoid common launch issues associated with a machine learning approach to recommendations. Normally, the sample system cannot be learned until the user rated many items. Knowledge-based Recommender system avoids this issue because the recommendations do not depend on the user's rating. Also, recommendations do not depend on user preferences, so there is no need to collect information about a particular user. Due to these factors, knowledge-based systems are valuable as stand-alone systems and are considered to complement other types of recommender systems. The main drawback of a knowledge-based recommender system is the potential bottleneck of knowledge acquisition caused by the explicit definition of recommended knowledge. Knowledge acquisition is the process of building the rules and a requirement required for a knowledge-based system and is achieved by acquiring knowledge through knowledge acquisition of rules, objects, and frame-based ontology and frame-based ontology. Bateson's theory has been used to guide the process of further learning of knowledge [22].

2.3. Hybrid method

Hybrid recommender systems combine two or more approaches for better performance. Its main goal is to eliminate each shortcoming. For example, collaborative filtering techniques have issues with a new item and new users' cold starts problems. For example, items that have not been rated yet cannot be recommended. This does not limit the content-based approach, as new item forecasts are usually based on readily available descriptions (characteristics). Based on two (or more) basic recommendations, several possibilities for combining them to create a new hybrid system have been proposed [22].

2.3.1. Hybrid filtering strategies

A. Feature Augmentation

This method is used to generate a rating for an item and then incorporate this information into the processing of the next recommendation method. The new feature of each item is generated by augmentation using the recommended logic of the serving domain. Feature augmentations are used when you have a well-developed key recommended component and need to add additional knowledge features or sources. In contrast to the cascading model, the augmentation hybrid method includes the output attributes of the first recommender in the feature used by the second recommender [20].

B. Mixed

This method is useful when many recommendations are needed at the same time. In the mixed hybrid method, the recommended components for that component are displayed side by side in the integrated list. This hybridization method does not attempt to integrate evidence among recommenders. Combining multiple independent lists is a difficult process this way. Standard techniques include merging based on predictive scores or trust of recommenders [17, 21].

C. Switching

The switching method selects a recommender from the component. For another user/profile you can choose a different system. For example, if content-based filtering cannot provide a high level of reliability and accurate recommendations, another method, such as a collaboration process is attempted. This technique does not avoid the entire problem that recommender systems have (that is, startup problems). This hybridization process assumes that there are reliable criteria for the decision to switch. As soon as the decision to switch is made, the other components that are not selected play no role in the recommended process on the left [16, 21].

D. Feature Combination

Feature combination enables the combination of one approach’s balancing features, e.g. CF recommender system, interested in an algorithm intended to process data with another method (e.g. CBF recommender system). Collaborative merging with content is achieved by treating the collaboration information as extra feature data linked to each model and using a content-based approach for this constructed dataset. This approach allows the system to examine aggregated data completely data-independent, making the system less sensitive to the number of users who rated the item [17].

E. Cascade

This technique is an organized procedure used to form a strictly hierarchical hybrid, in which a low priority weak approach cannot negate the decision of a high or strong approach, but rather can improve them. Low-priority recommendations are used to break the relationship between high-priority and high-priority evaluations. The lower priority method is not used for items that are already well distinguished in the first method. In addition, it is not used for items with a low rating, there for the item will not be recommended. Cascade technique is noise-tolerant in the operation of low-priority approach because they can only improve the rating and not the other way around [18, 19].

F. Meta-Level

The meta-level approach utilizes an output model learned from the recommender, another input. This method is different from the extension. Feature hybrids use the general properties of one trained model as input to another model, while meta-level hybrids use the entire trained model as input. Recommender does not work with raw profile data. Deriving a meta-level hybrid from a pair of recommendations provided is not always easy. This is not achievable with all recommended techniques, as the contributing recommender needs to generate a model that will be used as input by the actual recommender. The advantage of this technique is that the trained model displays a condensed representation of the user's preference. The collaborative approach makes it easier to use this compact representation than raw rating data [19, 21].

G. Weighted

This type of hybrid recommender approach aggregates the results of all joint recommendations approaches and calculates the recommended item/value score. A method of linearly combining multiple recommended ratings is used. The system initially gives the same weights to all recommenders and gradually adjusts the weights regardless of whether the user's rating predictions are checked. Although, this model implicitly suppose that the relative values of the individual methods are uniform across the possible items. This is not always factual to make fact decisions [18, 22].

In our work, the hybrid approach is used having both CF and demographic filtering. The CF is used to filter books and rated by similar users to estimate the future of the user. The second one, demographic filtering was used to cluster users according to their demographic information. And finally, a weighted hybrid approach was used to combine the result.

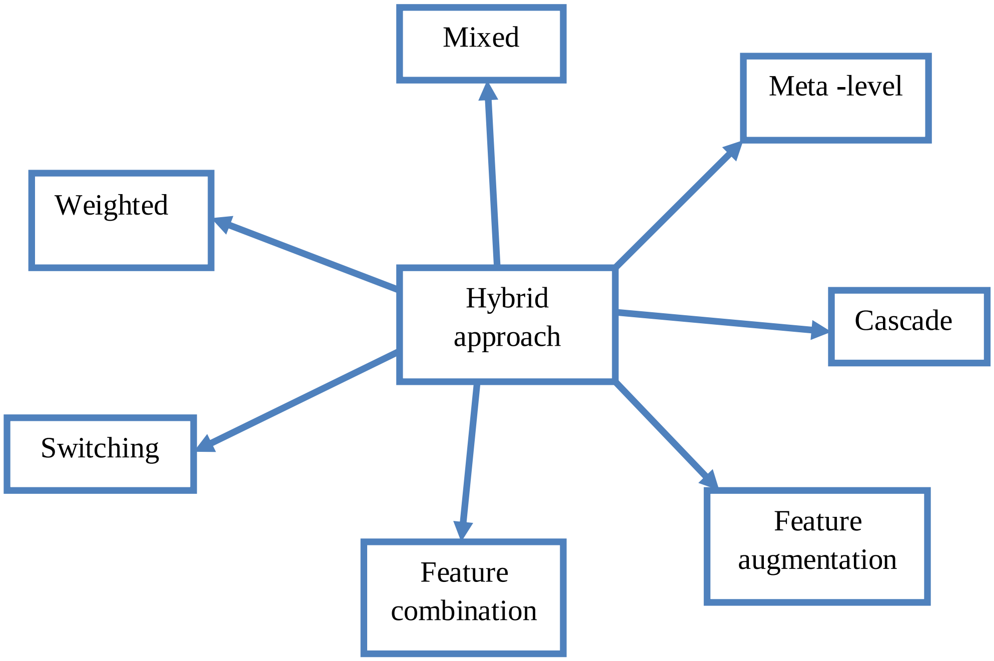


Fig 2. 2 hybrid filtering strategies.

2.4. Common Approaches of book recommender system

The usual approaches used in the book Recommendation System are CBF, CF and HF (That is, a combination of two or more approaches). The details of these techniques were discussed.

Content-Based Filtering (CBF):- is the common technique of the book recommendation system which is based on the properties of the book and tries to suggest books that are similar to those a given user have liked to read or rated in the past. This method uses the current user's rating history of previously used books to find the current user's preferences for the book. The similarity of books is determined by measuring the similarity of their properties. Therefore, this type of filtering method does not rely on other users' rating data records to generate preferences for the current user. CBF is based upon item properties. For instance, if a user buys a book on amazon.com using an RS, the user has another favorite for buying a book from an online bookstore that contains the same or similar keyword information for the book [13].

Collaborative filtering:-The other popular technique which is used in our study is Collaborative filtering. CF Recommender System recommends items to the user that people with similar interests and preferences have liked in the past. In this approach, the recommender system uses collaborative filtering to suggest items to the user and search for the user's peers i.e., a group of users who have similar interests in the item. Next, only the item that the user's peers prefer most is suggested [9].

Therefore, CF is an approach that aspires to learn user interest and make recommendations based on user and group data. This is a filtering technique that complements CBF (such as keyword-based searches). Perhaps the most commonly used application for collaborative filtering is amazon.com. It uses, for example, the user's previous purchasing history to make new item recommendations.

Additionally, CF is widely used as a type of personalized recommendation method in many domains [23, 24] and it provides some prominent advantages to information filtering:

1. the capability to filter items whose content is not easily analyzed by automated processes;

2. the ability to provide serendipitous recommendations; and

3. Support for social factors by taking into account the interests of like-minded users [24]

Regardless of the overall advantages of the collaborative filtering approach, it has some issues such as cold start, data sparsity, and scalability. These issues have a serious impact on user preference. With collaborative filtering, items are recommended based on user rating. So, a database of user past preferences should be obtainable. But, the database is for all time extremely sparse. That is, the user only rates a small number of items. So far, many researchers have focused on predictive accuracy and proposed several solutions.

The other issue of CF is the cold-start problem which occurs when there is no information about users and items. There are two main aspects to this: new users and new items. Before a recommender system can make a reliable recommendation to a user, it is probably necessary to know the user's preferences/interests from a sufficient number of behaviour records. i.e., Review or log archive [23]. Our study examines ways to address this cold start problem.

Hybrid: - The third popular approach in book recommender is hybrid, which was used in this study that includes both collaborative filtering and demographic filtering. The individual recommendation approach is not suitable for making good and accurate recommendations. Therefore, a hybrid recommendation system has emerged to overcome the shortcoming of the CF approach. The reason for choosing the hybrid approach is that the interaction of these two methods improves the performance of the system and the ability to serve the interests of the user.

For example, these systems are based on a combination of the benefits of multiple traditional approaches to generating recommendations, a collaborative filter approach with a content-based approach [25]. An easy way to take advantage of content-based recommendations is to combine both methods to get two separately evaluated recommendations and reach the final list that merges the two results. The two predictions using the weighted average are combined to increase the weight of the collaboration component as the number of users accessing the item increases. The following figure shows how Content-based filtering, Collaborative filtering, and hybrid filtering recommend items to specific users.

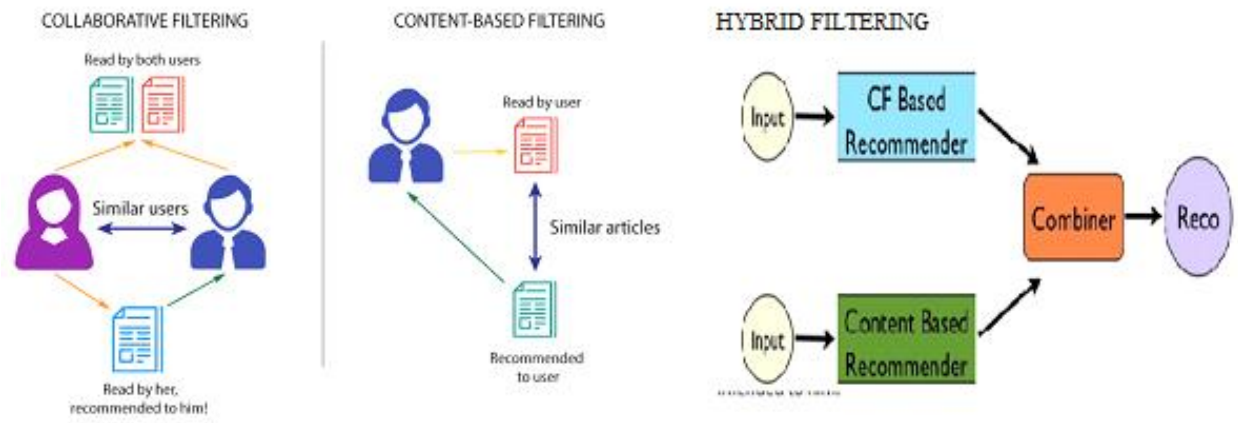


Fig 2. 3, Architecture for CBF, CF and HF

2.2 Clustering Algorithm

This is a technique that has been used to group data based on their closeness to each other to solve the time complexity and space issues during the search process. Clustering is a procedure or technique used to group data based on similarity to decrease the difficulty of query retrieval and the memory used to store the required data. The purpose of clustering is to reduce the time it takes to retrieve large amounts of data and reduce the amount of disk space used to process requests. There are two types of clustering; partitioning and hierarchical clustering [27].

2.4.1 Partitioning clustering

Partitioning clustering is a method in which the user must determine the number of clusters to relocate each object space, starting with the first partitioning. In this method of clustering, the process needs many times to achieve a better result by using all possible partitions [27]. Because of this, it is not easy to cluster the given large data sets. The most common type is the K-Means and it is discussed as the following.

2.4.2 K-means algorithm

This algorithm aims to classify a set of n contexts into k clusters based on their closeness to the center of the cluster. Closeness to the center of the cluster is measured using the Euclidean distance algorithm. K-means is an iterative clustering algorithm that moves items between groups of clusters until the desired setting is achieved. A high degree of similarity is achieved between sensations within a cluster, while a high degree of dissimilarity is achieved between sensations within different clusters [28].

2.4.3 Hierarchical clustering

The other common and popular clustering algorithm used to cluster the big data for the effectiveness of searching is Hierarchical clustering. It is the methods that create the clustering by segmenting the given data sets in a top-down or bottom-up way [27].

A. Agglomerative hierarchical clustering

The assumption of this method is initially started by representing each object in its cluster and then consecutively merging until the preferred cluster result is achieved. This is called the bottom-up way of clustering.

B. Divisive hierarchical clustering

In divisive hierarchical clustering first, all objects is considered clusters, and each cluster is divided into sub-clusters one after the other until the desired result is achieved**.** The top-down way of clustering which is starts from the assumption of one cluster into many clustering. The merging or division of hierarchical clustering is done based on the chosen similarity measures. Therefore, in our work, we use a hierarchical agglomerative approach to cluster the new demographic information of the users used as input data, and then the new users are most similar in the dendrogram, Connected to the cluster. However, the method of clustering used by this study differs from them. Usually, clustering is performed while mining the data. But the method proposed in our work is to cluster data while structuring the data model itself. It is used as a data structure for collecting data from the time of implementation and not after collecting user data. So the hierarchical clustering method proposed here is for data modelling and data management during storage. This is the upcoming research trend which is otherwise termed attribute clustering to cluster the similar attributes in one cluster of the given data [29].

2.5 Challenges of recommender system

The recommendation system by itself has many challenges to provide better performance. Those challenges are under research to be solved by many researchers using different algorithms. The most common challenges of recommendation systems are sparsity problem, cold start problem, scalability problem and over-specialization problem as studied in a survey of [30], [31].

Sparsity is the problem that influences the accuracy of the system. It is caused by a lack of users rating data. Since many users are not willing to rate the items they accessed this problem will occur. Without rating data, it is hard to know the user's history for future recommendations.

Cold start problem: is the common problem in recommendation system which happens because of lack of information of users need and the items needed to be recommended by the system.

Scalability: The problem of many data to be processed while the recommendation is done by the system. If the data used to provide a recommendation is huge like in the commerce system which deals with many users and many items with their rating history.

Over-specialization: Over-specialization is the problem in the items based recommendation for recommending the item based on their similarity. If more items are similar, then the items to be recommended from this group of items will become more and more which follows over-specialization.

The challenges listed out above are the main ones in the recommendation system and many studies have been done to fix them even though still they are not complete. In our research, we propose a solution to fix the cold start problem to overcome the problem of new users. Since new users have no history in the system, most systems will not recommend the related book for the users. So, the readers are going to read the book they will not like to read.

Summery

This chapter contains explanations of the basic concepts of literature or research areas. We presented and discussed the basic concepts of our field of study and some related approaches in our work. The basic concepts of the recommendation system in general and the book recommendation at a specific level with their techniques and each component of our works overviews are explained in detail. Some of the discussed concepts are summarized as follows: Recommendation system is recommending or suggesting the users save users from providing overwhelmed information depending on the history of users and the information or documents accessed by users. Therefore, the item needs to be known to the system and also needs the user's data to predict the user's interests. The system then calculates the percentage of items offered to the user from the user's past. Recommender systems take several approaches to provide the best results in the form of recommendations to those who search for resources from the system. Currently, collaborative filtering (CF), content-based filtering (CBF), knowledge-based filtering (KBF), and demographic filtering (DF), utility-based filtering (UBF), and hybrid filtering (HF) are the most common methods. Book Recommendation system is one domain of recommendation system used to suggest the book readers in online according to the interest of book readers from available books which are released daily in the world through the Internet. The book Recommendation used to provide the only interesting book with users from a huge source of book to the readers which are specifically needed by them by searching the related book for readers based on user’s profile history.

CHAPTER THREE

3.0. RELATED WORK

3.1 Introduction

This chapter describes various researchers in recommender systems. Researchers have focused on new user cold start issues as a major issue in recommender systems, and much research has been done to address new user cold-start issues to achieve effective performance.

3.2 Book recommender system review

Many recommender systems have been proposed to recommend books in the literature. Alie et al. [34] recommend books using different information filtering approaches, depending on the context and domain in which these systems are developed. For example, LIBRA [2014] is a CBF book recommendation system that uses book information from the Amazon.com website. Similarly, various studies have been conducted internationally and locally to consider book recommender systems. [33] Recommender systems explain that they are used to access relevant articles and information through personalized suggestions based on the user's previous interests and likes. These systems are used in a variety of areas such as products, videos, photos, articles, news, and books.

Sohail et al. [35] paper has implemented a fuzzy linguistic quantifier a soft computing-based fuzzy-based aggregation operator to calculate the aggregate score of the books. This paper has extracted features of books from online reviews and categorized them into positive and negative, OWA is used to calculate the final score and recommend the Top N books. The method was evaluated based on precision and the performance was promising. Zhang et al. [36] Implemented the Chinese library classification approach to the personal endorsement of books in the Chinese library. The system takes user choices into account and makes personal recommendations accordingly. Evaluated using perception, recall, and f-measure, demonstrating improved performance compared to traditional recommended approaches.

Sohail et al. [33] have extracted user feedback by extracting features from online website reviews for books recommended by experts in this field. After the feature is extracted, they are positive, negative, and reciprocally categorized, scores are assigned to each feature, and finally, the final score two of each book is calculated based on the review. Then a recommendation is generated for the user.

A hybrid book recommender system based on the Table of Contents (ToC) and a study conducted at the University of Peshawar using Association Rules Mining [39] shows that the information filtering method primarily provides relevant results that help users meet their information needs and reduce information overload and cognitive stress. In the end, he revealed that different recommender systems are being developed in different fields, but there is a lack of systems that are dedicated to books and can recommend books based on their content.

This study proposes a collaborative demographic filtering approach to book recommendations from new users to recommend new users who do not have historical data in their system. The proposed system uses a combination of both user rating data and demographic data. Demographic data sent by the user to the system is used to enable recommendations for new users who do not have a history (rating) on the system. This proposed model attempts to solve new user problems by grouping new users with existing users based on the demographics registered in the system.

Table 3. 1 Summaries of related works

Author(s) (Year)	Problem/gap	Methodology/approach	Key findings	Conclusion remarks
Book recommender review				
Chandak, Girase and Mukhopadhyay , (2015)	Existing recommendation techniques use these user and item features for a recommendation but they are not	Hybrid	Describes an effective hybrid book recommendation technique that uses ontology for user profiling to improve system efficiency.	An RS that supports a large amount of data and processing.
Jomsri, (2018)	Further differentiators need to be identified to optimize these techniques**.**	Mining	The FUCL mining method is suitable for library recommendation tools and has a higher accuracy score than other methods.	Users in the same department borrow books in the same category. That is, users are interested in the topic of the same book.
Ali, Khusro and Ullah, (2016)	The existing book recommendations face some challenges in making related book recommendations. Most of them do not consider the contents of the book at a	Hybrid	A hybrid book recommendation system that recommends books using a book table of contents (TOC) and mining association rules and similar user feedback.	Information filtering techniques have been used primarily to develop recommended systems to mitigate information overload problems.

	deeper level.			
Sang Nguyen, (2019)	Processing data, choosing the right data characteristics, and how to classify them are always challenges in determining the performance of recommender systems	CF	Naive Bayes is perfect for book recommendations with acceptable processing time and accuracy.	Data processing, features, and classifier selection to build an efficient book recommender system
Anwar, Siddiqui and Sohail, (2020)	Machine learning technique Performed with a book recommendation**.**	survey	Evaluation, a method used to test **system** performance and **prospects**	These systems are extremely helpful in decreasing information overload and users preference
Uko, O. and O., (2018)	With the help of recommender systems, relevant information needs to be filtered, prioritized and served efficiently**.**	Experiment	Design and development of recommender models using object-oriented analysis and design methodology (OOADM)	The rate and scalability of the book recommendations have been improved by recording performance

As shown in the table above, various researchers working on recommender systems typically classify the system as collaboration, content, demographics, and/or knowledge bases. Though, hybrid systems combine two or more algorithms to improve recommendations and overcome the limitations of each approach.

CHAPTER FOUR

4. Proposed model

In this chapter, we describe the problem are attempting to address as well as our approaches to resolving the problem and how we will complete each aim to attain the overall goal. The data provided in the proposed system demonstrates how the system works, and the methods we utilized to ensure proper flow were defined and addressed as follows.

4.1 Approaches of the proposed system

Our work proposes Collaborative Filtering approaches with user demographic information, which we present. By addressing the problem of new user cold start, our strategy in our work aims to alleviate the weaknesses of collaborative approaches to provide active recommendations to book readers. The Collaborative Filtering technique detects similarities between users and books and groups them together.

4.1.1 Collaborative Filtering (CF)

The data of the book to be recommended and the data of users who require a recommendation are known challenges in the recommendation system. Users' lack of information and the requirement for a new item are both significant obstacles in the recommendation system. CF is one of the most prominent ways in this system, and numerous academics have modified it to address these common issues. Collaborative filtering is one of the most widely used approaches in personalized recommender systems, providing suggestions to users with similar review histories, assuming that they will have similar preferences in the future. This indicates that the Collaborative filtering algorithm suggests items based on historical data from people with similar likes. To achieve this personalization, recommender systems need to maintain specific information about user preferences, called user profiles [27]. In the proposed book recommender system collaborative filtering method takes into account the assumption that users who rate or read a book will have a book of the same taste for future books.

In the case of cold-start a new user, the problem occurs when the new user is introduced to the system and lacks background information regarding the user, including the item/product he/she are looking for.

In our case, the new user problem occurs with the lack of rating data for the book in the system. The collaborative Filtering needs user similarity for predicting the books to be recommended based on the books rated by similar users.

To find out the similar users firstly, we need to have the users demographic since our problem targets recommendations for the user who had no history in the system. In the majority of present recommender systems, demographic data like user age, profession, location and sex are used as users' attributes to calculate similarity among users. We applied the CF approach in our works to overcome the problem of a cold-start new user. The approaches consider user similarity based on the history of books that have been rated in the past and stored in the rating database. So, the new users who have no history in the system should be added to the active users. Active users mean the users that have rated books and the history information in the system. To add the new users, we need to have the information used to add to the active users based on the similarity. Since we have the assumption of the users with similar demographic information will have a similar taste of books, we considered the demographic information as the information used to calculate the similarity among the existing and the new users. So, their similarity is done by their demographic information clustering.

The clustering of the user's dataset based on their demographic information is done as it is described in the data pre-processing section. Then, after the new user joined the system group is identified the books rated in this cluster is filtered and the filtered book is predicted to the new user. But the generation of new for these users should consider the rating value given by the active users. This approach needs both the online process to cluster the new user into the active users and the offline process to fetch books rated and stored. In our proposed work, we applied both model-based and memory-based algorithms. The model-based is applied while clustering the users online by learning the user's neighbors from the system and the memory-based is used while our system filters the books from the memory.

i. Memory-based CF

This technique uses the data stored on the memory from the database and processes them to recommend the item for the user considered to this rated item. In our case, we applied this for the rated books which are stored and we predict these books for the more similar users based on the history of rating the users had in the dataset. Memory-based collaborative filtering is typically divided into user-based and item-based. We used **user-based** collaborative filtering for **the** proposed model.

ii. Model-based CF

Another method of CF in recommendation system which learns the user interest while processing to predict users interest or the books recommended for the user. The users clustering process in our proposed system is done by this model-based since the system learns the user appropriate groups online after accepting user demographic information from the user system. In the proposed algorithm, the new user has no history in the system, so this model-based CF requires a different user history. We added user demographic information in addition to the rating data of existing users. The users are clustered based on their demographic data and the existing clustered users have the rating value. Therefore, you can predict new user ratings based on existing user ratings.

4.2 Proposed System Architecture

The architecture presents the model and the algorithm used to achieve the task to be done in the work. Since the problem is to solve the new user recommendation the work includes many components. As discussed in chapter two, the new user problem is the universal challenge in RS. New user challenges are defined as issues that occur when there is no information in a transaction in a recommender system. Transactions in recommender systems are interactions between users and recommended products or recommended information. In our case, most book readers need only books that are more interesting to them. If the reader is unfamiliar with the system, they will face cold start issues due to lack of preference information. Therefore, entering user information is one way to overcome this problem by feeding the user's demographic information. The entire proposed system architecture is shown in Figure 4.1, and each key component is described in the next section.

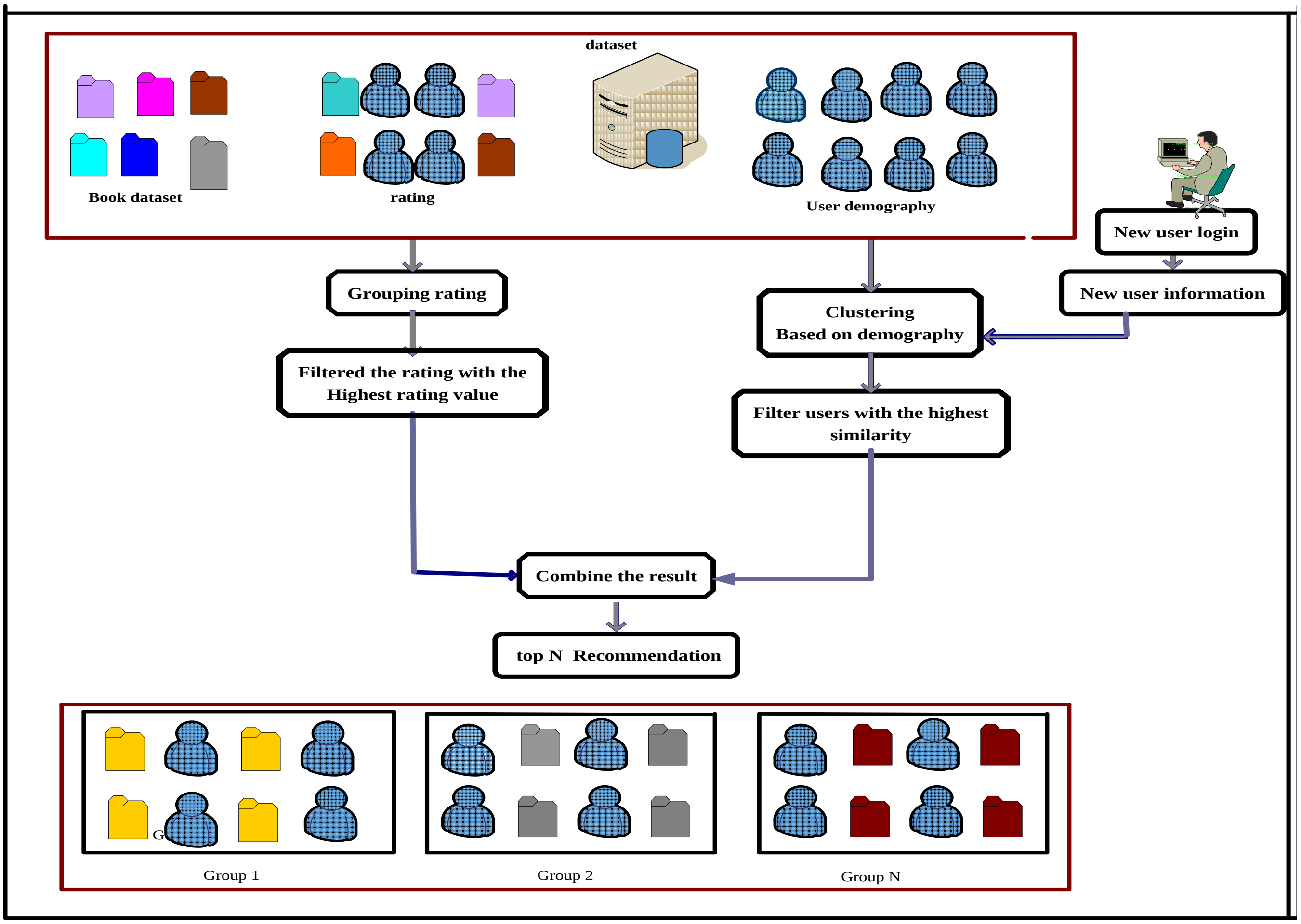


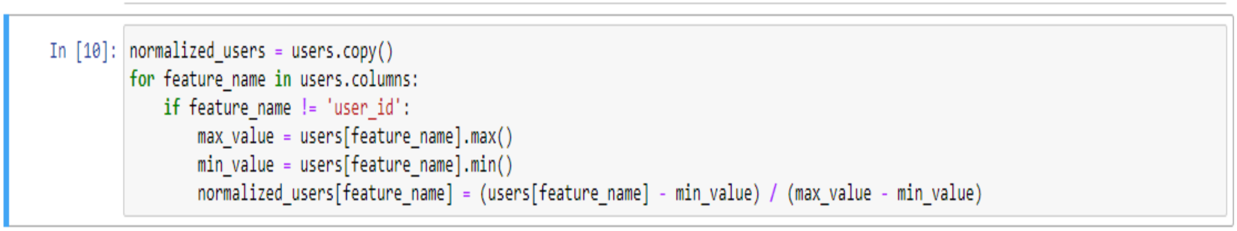
Fig 4. 1 proposed recommender model.

4.2.1 Components of Proposed System

The pre-processing of the dataset, accepting the new user demographic information and identifying the user cluster and finally recommending the interesting books for the users are the major tasks to be done in the proposed model. These are discussed in more detail in the next section, why they are needed and how they are functional by supporting them with the algorithm.

4.2.1.1 Pre-processing Data set

The dataset we used for evaluating our work is huge and we need to **pre-process** by using a K-means clustering algorithm used for pre-processing. We used the user demographic data, book and rating data of book by users. The number of users we used in this data is not easy to process and it is difficult to search compare the similarity of each user with all existing users. So, we applied the clustering algorithm to find similar users based on their demographic data. We used the K-Means clustering algorithm to pre-process the existing user dataset by using the similarity of user demographic data in the dataset. For the similarity, we selected the appropriate attributes used to cluster the data. The second dataset we used in our work is book data. The third and last data set we used for our work is the rating data achieved by the interaction of the book and the readers. And we group them based on the rate values.



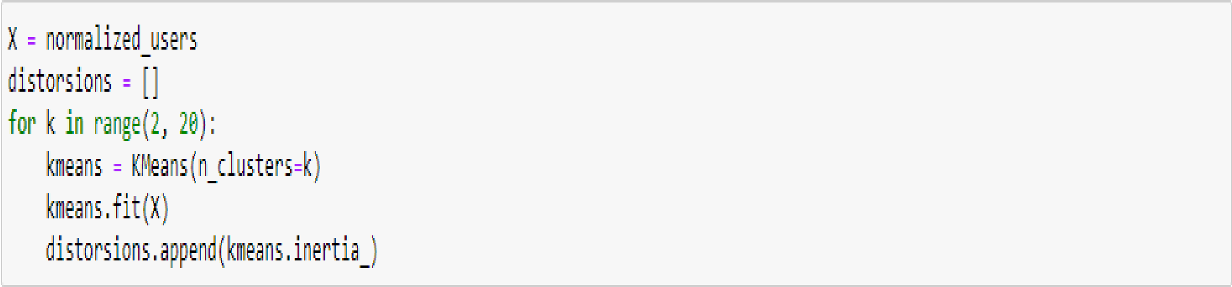
```
In [10]: normalized_users = users.copy()
         for feature_name in users.columns:
             if feature_name != 'user_id':
                 max_value = users[feature_name].max()
                 min_value = users[feature_name].min()
                 normalized_users[feature_name] = (users[feature_name] - min_value) / (max_value - min_value)
```

Fig 4. 2 Python code to normalize the dataset.

A. **Clustering Users**

We clustered the existing user dataset by grouping users based on the attributes we selected from user demographic information to cluster them. The existing users are clustered to their appropriate group based on their similarity according to their age, sex and occupation group. We clustered users based on their occupation, sex and age. We selected these attribute since we have the assumption of the person with a similar occupation have a similar interest in the book because people want to get information about their occupation every time. In addition to this occupation data, we cluster users according to their sex since males and females have different interests in a book for their personal life. For example, female users want the book of fashion for females and many books related to female users. In addition to this, age also influences user interest as young users and old ones have a different interest in the book. This clustering is important and we use it when we search similar users for the newly registered users to reduce search complexity to find the specific user groups needed to be recommended. Since we are recommending new users, it is better to know the user's group by identifying the registered information about users.

We always check the similar groups for the registered new users based on age group, gender and their occupation online process. But the existing users are clustered to their appropriate group based on their similarity according to their occupation, age group and gender offline before the new user registration process. When a new user registers, the process of searching for the appropriate cluster of a user will continue and if a cluster is found the recommendation will be processed using the history of the existing user similar to the registered new users. But if the new user couldn’t get the exact group, a new group for the users is created in our system. In addition, the recommendation provided for these new users is the popular book filtered. The algorithm for user clustering is as follows.



```
X = normalized_users
distorsions = []
for k in range(2, 20):
    kmeans = KMeans(n_clusters=k)
    kmeans.fit(X)
    distorsions.append(kmeans.inertia_)
```

Fig 4. 3 python code for K-means clustering

4.3 Rating dataset

The other data set we need to cluster is the rating data of books rated by users. This data set contains the book rated by users with the user ID and book ISBN. In addition to this, the rating data contains a rate value. In the proposed system, the new user registered for the system looks for a book and the system provide a book requested by a user based on their rate values. So, the system needs to process the rating data which is huge. The processing of this huge data takes many time and memory which follows the scalability problem. Therefore, this is an important issue to consider in the study and we used the grouping data scheme for this dataset. We apply the method to group the rating dataset. The grouping data methods for this dataset are done based on the rate value of the book given by the users who read and rated the book. Since the grouping considers the values of the book rating given by the users and we need a book with more rating values this also reduce the book those have fewer rating values in our works. After we grouped them we have 4 groups of books. Those are above average book rating data which contains book rated by a user with the 6-10 rating value, average rating data with 5 rating value, below rating data with 1-4 rating value and visited rating data with no rating value but visited by the readers.

4.3.1 Registering New User

We focused on user data to achieve the main objectives of this study. In this study, our proposed system is mainly used with user information. The user information we used for our study is the demographic information data. The user we need to recommend in this study is the new users who have no information from the system to provide the recommendation. Registering user enables us to collect the new user demographic information. Since our system needs the user demographic information for recommendations, we submit new user demographic data to the system by using the user interface we prepared. These collected data of new users are recorded on the user dataset and the users with the most similar demographic attributes are clustered under a similar cluster. The clustered users also need to be registered for their groups by the system. This data is used for our assumption that the users with the most similar demographic data have a common interest in the book. So, any new users need to be registered for the system to have a recommendation of the book.

4.3.2 Clustering New Users

The registered user groups should be identified according to user similarity and the data are registered to the appropriate cluster or similar groups. This identification of new users is done based on demographic attributes. If the registered new users most attributes are similar to existing users' demographic data, the users will belong to that clustered existing users. Then the system adds the user demographic data into his/her respective group. As explained in the section above, users grouped among users who are similar by occupation and gender may be many users and we need to reduce the number of users under the selected clusters. Filter this information to reduce the number of users, assuming that users of similar ages have similar interests in the books they are reading.

Firstly, we grouped users into five age groups. The first group with the age range of 17-25, the second group with the range of 26-35, the third one with a range of 36-45, the fourth one is 46-60 and the last group with a range above 60. From the four age groups, the new users will be in one group and the book rated by these users in this group is fetched from the rating database. The following graph shows the age group distribution on our model.

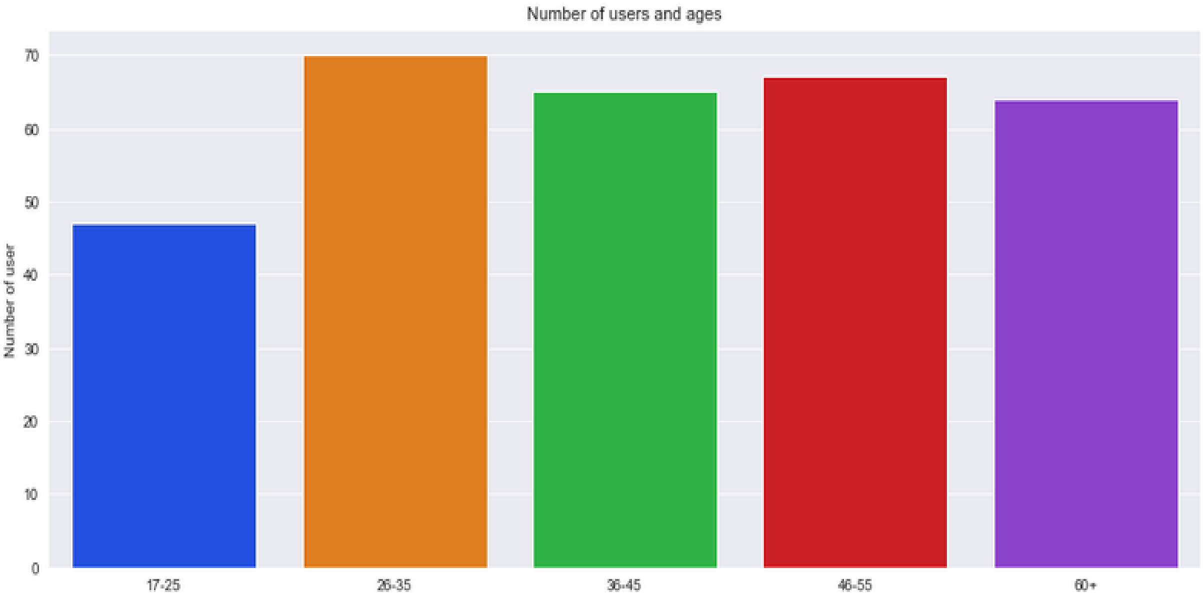


Fig 4.4 1 age group distribution on our model

4.4 Retrieving Rated Book

Since the recommendation should have to provide the book, retrieval of a book should be done by the system using the proposed model. So, that the filtered users based on age group from the particular cluster get the rated book. The rated book by filtered users should be fetched and the process of priority consideration is applied to them before predicting to the newly registered user. Retrieving books is done using collaborative filtering approaches.

4.4.1 filtering popular

The popular books assumption in this work is the books with the highest rating value. Then, we check their frequency or the number of occurrences since the book frequently occurred is the books rated by many users and it is popular with many readers. Finally, we retrieve these books by checking their rating value and retrieving all with the highest rating value.

4.4.2 Filtering book by collaborative filtering approach

Book rated by the user under one group according to their demographic similarity is filtered. Since the rated books are clustered into five different groups, the system should check from all groups and follow the priority to return the book. The book from the highest rating value should get the priority if it fulfills the number of books to be generated for the readers. If not the next group is checked and also it should fulfil the determined number of books to be retrieved from this group.

4.5 Combining Results

To overcome the separated approach of book recommendations we applied collaborative filtering with demographic data. The approach is done by combining the different results using the weighted approach obtained by the separated approaches we used in our study. The combination of the result of both books is recommended based on rating prediction and demographic filter based on the frequency of the book rated by many readers and the rating values given by the readers. The proposed approach is done by combining the results obtained by both predicting and demographic to be recommended after ranking.

4.6 Generating Top Recommended Book

It is the stages to provide the books which are selected to be recommended by the system for the readers is done by top values method. The top N book recommendation is done by the ranking algorithm we developed for the book recommendation system. Since the information our work recommends is the book we need to consider the highest rated value and the popularity of the book by readers. The book to be recommended is retrieved through both the two approaches collaborative filtering and demographic approach generated based on popularity. Since each of the approaches we have used has its ranking methods for the book to be generated and the results obtained are generated by the ranking algorithm of each of the approaches used. And the results obtained in each approach are combined by aggregating their results. Finally, we generate the book to be recommended.

Summary

We have proposed a new model used to recommend a book to new users. We have used the dataset of the existing user and books with the rated value to cluster the users and books according to their similarity and finally, we recommend predicting the new user based on the similar use of the demographic information with existing users. We identified each of the components of our model and with the architecture, we proposed and we put the entire algorithm of the proposed. The scalability problem in the book recommendation is fixed by using clustering methods. It is done by clustering the dataset used in this study. The user should register to get the recommendation system by submitting the demographic information into the system through the user interface. Then, the system clusters the new user registered into the appropriate cluster based on the demographic information submitted by a user. Users are filtered from a cluster of users based on the age group of the registered users and get books rated by the filtered users. These books are predicted to new users through a collaborative filtering approach. Popular books are retrieved based on the frequency of books rated by users and the rate value.

CHAPTER FIVE

5. The result, Discussion and Evaluation

5.0. Overview

In this section, we present the implemented model and experiment conducted in this study with the result obtained for evaluating the performance of the study regarding the problems we studied and comparing with other previous works done by other researchers. To do so, we need to have an analysis of the data set that we used for implementing our model. The results generated by the implementation are discussed and the evaluations of the result obtained are done by using the metrics we used according to our problem dimensions.

5.1. Experimentation

In the next section, we discuss the dataset used for the experimentation, the tools used for implementation and the evaluation metrics used to measure the performance of the proposed system.

5.1.1. Implementation Tools

For the prototype of the proposed system, the development platform and programming language used are Windows environment and python anaconda respectively. Python is a high-level, interactive, object-oriented scripting language, which is designed to be very readable. English keywords are often used, but other languages use punctuation and have fewer syntactic dependencies than other languages.

5.1.2. Dataset

To overcome the new user problem, we have used the book crossing dataset. This contains three types of data set which are book dataset, user demographic information and rating dataset. The data source for our model was found online at [8] Websites.

This dataset contains user data, book data and ratings. User data contains user demography namely gender, age and occupation. Book datasets include ISBN, book title, Book Author, Year of Publication, Image-URLs, Image-URLM, Image-URL, and Publisher. The last one is rating, which consists of user-id, book ISBN and rating value which is either explicit or expressed on a scale of 1-10 (higher values mean higher ratings) or implicitly expressed as 0.

5.1.2.1 User demographics dataset

It is data of users which includes user age, user location, user ID, user occupation and sex. It is important in our work since our work is purposely based on the new user problem and we have to have their demographic information. As mentioned above, our objective is to solve the new user problem which happens by the lack of information in the system. So, we need to have personal information to predict what will be their interesting books to be read based on the similarity of them with existing users according to their personal or demographic information. Since we assumed that some users may have similar interests with their age groups or their similar occupation. The dataset contains three attributes; those are User ID, Age, sex and occupation. They are described by their function in our system and their data types as follows.

A. User ID

It is the primary key in the table and is used to identify each of the users in the system and it is given by the system. Its data type is character. It is important to know the user who rated the book in the rating data using the user ID as a foreign key.

B. Age

For identifying the interest of readers in a similar age group since, users in the similar group may have a similar interest in our assumption and it is also analyzed in the next section, why it is used to cluster the users. It is the integer value.

C. Sex

The sex of the user is also another data that will influence the interest of book categories since female and male interest is different. This will be analyzed in the next section with reasoning. Generally, user demographic information is the information used for finding similar users and clusters them according to their demographic information similarity for this work.

D. Occupation

This is the information that is used to identify the profession or the jobs of the user to suggest the book related to his/her occupations. The users in the dataset are clustered based on these attributes and, why it is selected will be discussed in the Attribute analysis section. It is a text data type.

5.1.2.2 Book dataset

The book dataset contains the book metadata which includes ISBN Book title, author, and publication year, Image URL-S, Image URL-M, Image URL-L and publisher.

5.1.2.3 Rating value data

Rating is the value given for some online items by the user of the system for commercial purposes and for reading or watching videos. In the proposed work, rating data is the dataset that holds the value of rating given by users for a specific book by existing users for an existing book in the dataset. The table of this dataset in our database consists of the user Id, the rating value which is 0 up to 10 and the book ISBN, which are discussed below.

User ID

The attributes are used to identify each user in the system and it is given by the system. Its data type is an integer. It is important to know the user who rated the book as a foreign key.

Rating value

Rating value is the value that is given by the user for each book and is the integer value between 0 and 10. It is the attribute used to cluster the rating data set to put the book that has related value in similar groups.

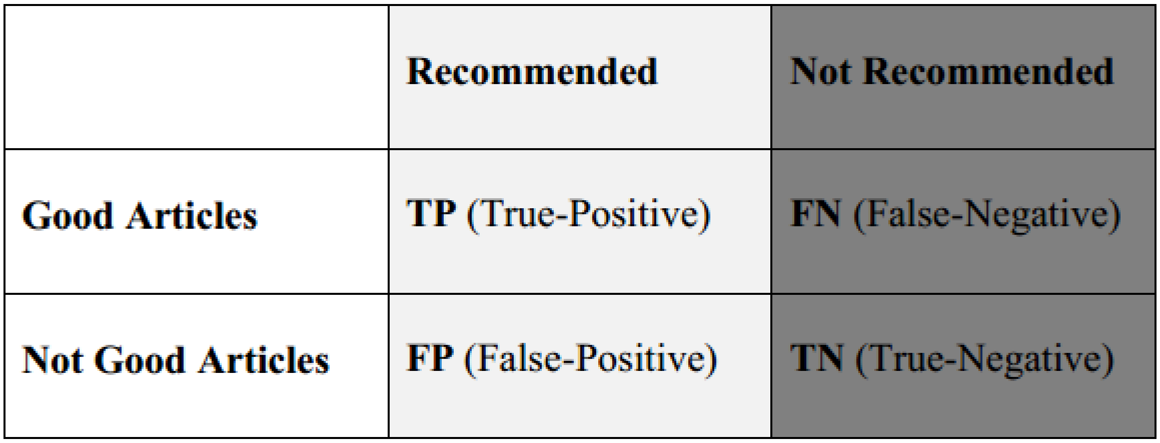
ISBN

The book rated will be identified by this attributes called ISBN it is the unique identifier of the book on the data since our system needs the average rating value of similar books in similar user groups. It is integer data types forwarded from books dataset.

5.2. Evaluation Metrics

To evaluate the performance of the proposed model, we depend on the objectives of our study and we selected the related metrics to our objectives. Since the main objectives in our study are to recommend the more related or interested book for new users we should have to evaluate the relatedness of the book with users. So, the popular and the most used metrics for any information retrieval to measure the relatedness or the interests are precision, recall and F-score. The following table shows the descriptions of all metrics with their relationships [39] Table.

Table 5. 1, descriptions of all metrics with their relationships



	Recommended	Not Recommended
Good Articles	**TP** (True-Positive)	**FN** (False-Negative)
Not Good Articles	**FP** (False-Positive)	**TN** (True-Negative)

A. Precision

Precision is the most performance measurement in both traditional information retrieval and the different RS and which measures the performance of the system that provides the information with the relatedness of the information retrieved. So, in our case, the book recommended is measured by how many books are correctly recommended out of all recommended books based on the new user interest. This result will be found by calculating using Equation 1, [40].

$$\text{precistion} = \frac{good\ Books\ Recommended}{All\ Books\ Recommended} \ldots \qquad \ldots (1)$$

Or $$\text{precistion} = \frac{TP}{TP+FP}$$

B. Recall

A recall is another measurement of Recommendations to measure the performance of many systems and it measures how many items are correctly retrieved in the information retrieving system or recommendation system. In our case, it measures the good recommended out of all good recommended items as described in Equation 2, [40].

$$\text{recall} = \frac{good\ Books\ Recommended}{All\ Good\ Books} \ldots \qquad \ldots (2)$$

$$\text{Or} \qquad \text{Recall} = \frac{TP}{TP+FN}$$

C. F-Score

F1-Score is the harmonic mean of the result of precision and recall. as it is formula is shown in Equation 3, [40].

$$\text{F-score} = \frac{2(good\ Books\ Recommended)}{2(Good\ Book\ Recommended)+Good\ Book+Not\ Good\ Books} \ldots \qquad \ldots (3)$$

$$\text{Or} \qquad \text{F} - \text{Score} = \frac{2TP}{2TP+FP+FN}$$

5.3. Experimentation process

The findings of this research are analyzed in the following way to show what was accomplished and how well our model answered the problem mentioned. The discussion and explanations of the acquired results will be presented in a figure or table format with additional clarifications. This paper discusses the performance outcomes we received in this investigation based on our measures. We used 23, 455 active users, representing a variety of occupations, both sex, and ages. These selected users are the users who rated 271,380 books and generated 1,048,576 rating data. The users were divided into 30 categories based on the 21 occupations. Based on the rating value ranges, the rating data were divided into four groups. Rate values of 1, 2, 3 and 4 are considered below average, rate values of 5,6, and 7 are considered average, rate values of 8,9 and 10 are considered above average, and Books with a rate value of 0 or not rated by the user are considered visited. The system generates books for the newly registered user and assigns them to the appropriate cluster; if the new user's cluster does not exist in the system, it is created and filled with data. Once people have registered, we employ a K-means clustering method to group them. When a new user joins a system and requests a recommendation, the system requests the user interface designed for the new user, and the system makes recommendations on the interfaces based on the information gathered from the users.

5.4 Experimentation for individual user similarity

For the individual users, we evaluated by taking 15 active users in the dataset. For preparing the testing dataset we used these 15 users rating data. Then we remove the book rated by these 15 users and we register each of these 15 users as a new user and we run our proposed model to recommend the book for the users and we compare the previous book rated by the user with these actual recommendations. Then, we calculate the precision, recall and F1-score values as the formula we discussed in the previous section. According to this experimentation, the result of the work is explained in the Table shown as 5.1.

Table 5. 2 Experimentation value based on each user similarity

User-id	**Precision**	**Recall**	**F1 score**
256925	0.563332	0.398733	0.573683
269855	0.621853	0.414014	0.490567
253685	0.697038	0.57001	0.48655
268522	0.551482	0.396948	0.49964
258647	0.641113	0.420381	0.490112
289654	0.71961	0.527674	0.409364
658397	0.720742	0.40401	0.641875
258469	1	0.505264	0.654099
248932	0.781231	0.424241	0.49111
258764	0.680756	0.429294	0.599555
256983	0.741482	0.49898	0.591
256893	0.655147	0.543039	0.506317
275862	0.583123	0.398278	0.49191
255486	0.561321	0.488536	0.400376
258665	0.60717	0.461384	0.50011
Average	**0.675856**	**0.45805**	**0.5217436**

From the evaluation results, we have the accuracy of the books to each user with the values of 67.58% of average Precision, 45.80% of average recall and 52.17% of the average of F1_score values as shown in the table.

5.4.1 Experimentation by user cluster based similarity

The other way we evaluated our performance is based on the user cluster. We took 5 user groups. Then, we recommend some of the new users similar to the cluster selected and Compare accuracy performance by comparing actual recommendations with those recommended for this cluster. According to these experiments, the results of the work are explained and shown in Table 5.3.

Table 5. 3 **Experimentation values for cluster based user similarity**

Cluster Number	**Precision**	**Recall**	**F1 score**
1	0.89986	0.409639	0.571429
6	0.96911	0.421053	0.585915
10	0.98988	0.408	0.569832
15	0.891304	0.366071	0.518987
21	0.92799	0.407	0.559832
Average	**0.9355198**	**0.4023526**	**0.56119**

Based on this experiment we have the accuracy values that performs the average precision of 93.55%, average recall of 40.23% and average F1-score of 56.11% and the average of the precision on this experiment and it is shown in table 5.2

5.5 Discussion

The result of our experiment in this work contains two experimentation ways and each has different values as discussed in the previous section. Finally, we analyzed that the recommendation performance is different as obtained in two ways of our evaluation methods. The accuracy result of the recommendation is different according to the similarity of the users with clustering-based and individual user similarity. According to the two experiments results, we analyzed that the more accurate recommendation is done for the users in the clustering which performs the precision of 93.55 %, Recall of 40.23% and F1-score of 56.11% rather than individual user recommendation which performs 67.58% of average Precision, 40.23% of average recall and 52.17% of the average of F1_score. This is done because the cluster based recommendation contains more related books regarding the users in that cluster more than the books recommended for individual user similarity. According to this value, the Recall in the individual performs less performance than in cluster-based. The reason behind this result is the finding of good recommendation result numbers from many users in the cluster. So, if the good books recommended are many then, the Recall value will become less.

5.6 Comparing Result Performance

The previous study result performance done for recommending books and to solve new user problems are analyzed as follows.

1. Ullah et.al [38], proposed A Hybrid Book Recommender System Based on the Table of Contents (ToC) and Association Rule Mining University of Peshawar, Peshawar. book content, item-item CF approaches are used to develop the model, and association rule mining to recommend more accurate and relevant books that meet the needs of book readers and mitigate cold-start issues for items. In the evaluation of their model, it is shown that the hybrid model outperforms CBF and CF. Then they had performed the 78% average value for precision, 71% average for Recall and average for 74% F-Score.
2. Sunitha and Adilakshmi [53], to address the cold start issue for new users, proposed a new approach that the user side in addition to the user-item rating matrix. User side information is retrieved from social networks. The result shows that the 0.08149 precision and 0.16655 recalls were achieved in both personality-based similarities and hybrid schemes.
3. Jomsri, P [36] uses book categories and book lending or FUCL technology to develop methods that recommend the best book for users, depending on users' faculty. It is based on the combined functionality of association rule mining. the evaluating their method archives92% precision value.

Table 5.4 Result comparison

Authors	**Performance metrics**		
	precision	**Recall**	**F1-score**
Ullah et.al [38]	**78**	**71**	**74**
Darvishy *et al.* [53]	**0.08149**	**0.16655**	**-**
Jomsri, P [36]	**92**	**-**	**-**
The proposed model	**0.9355198**	**0.4023526**	**0.56119**

Our study is different from the objectives and we evaluated based on our objectives. Since we have begun the study by different objectives the data set we used is not used in other studies of recommendation. Generally, the performance of our study performs the good recommendation accuracy.

Summary

This study discussed the dataset, tools we used for implementing the model stated and we discussed the performance evaluation of the results obtained. The results obtained are also discussed and shown in screenshot figures. The main concepts described in this chapter relate to the analysis of results using various metrics to assess the performance of the proposed system. The output of our work with the user interface designed is discussed and the performance of the work is also evaluated and explained in different tables and figures forms by discussing the metrics we used according to our objectives. Each of the metrics used is also discussed with their appropriate formulas. In addition, the result from the analysis, the dataset used and the obtained output by the algorithm proposed is discussed with the screenshot figures. The evaluation metrics used are discussed and the experimental result of them is also identified for all the way the system is evaluated. So, the performance of this study finds the good interims of precision, recall and F1-score.

CHAPTER SIX

6. Conclusion and Future Work

6.1 **Conclusion**

In this study, User Demographic Data with a collaborative filtering book Recommendation system is proposed. This recommender combines a collaborative approach with user demographic information. The collaborative filtering uses the user similarity to check the book rated with similar groups of users. User similarity is based on demographic information that the user logs in to the system at system startup**.** The user data such as age, sex and occupation and user id in the system helps to identify neighbor users and/ or similar users from existing users by using the K-means clustering algorithm to cluster the user neighbors. In addition to these approaches, the popular book mostly read by many users and rated with high rate value is filtered. Finally, the proposed approach combines the results and provides recommendations by ranking the highest rating value and recommending relevant books for the new users. In addition to this, the large data processing or the scalability problem is handled by using K-means clustering used in this study to cluster the book data and user demographic data.

Extensive experiments are conducted to evaluate the performance of the proposed schemes recommended in the book. Moreover, the performance is evaluated using Precision, Recall and F1-Score of information accuracy metrics. The proposed model produced efficient performance results in terms of precision, recall, and F1 Score of information accuracy metrics. The experiment results demonstrate that User Demographic Data with a collaborative filtering Recommendation system achieves satisfactory performance results. Moreover, the proposed model performed a Precision value of 67.55%, Recall value of 40.23% and F-Sore of 52.17% for individual user similarity experiments. And the user cluster based experiment performed a precision value of 93.55%, recall value of 40.23% and F-Score value of 56.11%.

In this proposed model, we include a User Demographic Data filtering approach towards recommending a relevant book for the new users, in addition to combining collaborative filtering with demographic data hybrid approach. It is indeed, there is a satisfactory performance improvement. However, the performance assessment should be conducted using a real dataset to take into account different considered scenarios, to conclude the comprehensive effectiveness of the proposed model.

6.2 Future Work

The effectiveness of a given book RS is determined by the features a given approach utilizes, either feature of the books itself or information about users, both implicit and explicit data about the user's online behaviours. In regards to information about users, one aspect which requires further investigation is combining two or more user approaches, to improve user similarity so that suggestion of relevant books for a new user would be better improved. Moreover, the performance assessment should be conducted using some real datasets taking into account different considered scenarios, due to the heterogeneity of users' behaviours. In addition, supervised machining learning approaches can be another line of future work to streamline the design of an effective recommender scheme to enhance the performance towards addressing new user cold-start problems.

Reference

[1] Ebadi, A. & Krzyzak, a Hybrid Multi-Criteria Hotel Recommender System Using Explicit and Implicit Feedbacks. Proceedings of 18th International Conference on Applied Science in Information Systems and Technology (ICASIST 2016), Amsterdam, hlm.

[2] Lu, J., Wu, D., Mao, M., Wang, W. & Zhang, G. Recommender System Application Developments: A Survey. Decision Support Systems 74(12-32.) 2015

[2] F. Xia, S. Member, N. Y. Asabere, H. Liu, Z. Chen, and W. Wang, "Socially-Aware Conference Participant Recommendation with Personality Traits," vol. 00, no. 0, pp. 1–12, 2014.

[3] G. Adomavicius and A. Tuzhilin, "Towards the Next Generation of Recommender Systems : A Survey of the State-of-the-Art and Possible Extensions," pp. 1–43.

[4] O. H. Embarak, "A method for solving the cold start problem in recommendation systems," in *Proceedings of 2011 International Conference on Innovations in Information Technology,* Abu Dhabi, United Arab Emirates, 2011, pp. 238-243.

[5] J. Basiri, A. Shakery, B. Moshiri, and M. Z. Hayat, "Alleviating the cold-start problem of recommender systems using a new hybrid approach," in *Proceedings of 2010 5th International Symposium on Telecommunications,* Tehran, Iran, 2010, pp. 962-967.

[6] C. Shahabi and Y. Chen, "Web Information Personalization: Challenges and Approaches," i*n Bianchi-Berthouze N. (eds) Databases in Networked Information Systems*, vol. 95, pp. 1–10, 2003.

[7] Gediminas Adomavicious and Alexander Tuzhilin, "Towards the Next Generations of Recommender Systems: A Survey of the State-of-the-Art and Possible Extension," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no 6, pp: 734 - 749, 2005.

[8] http://www2.informatik.uni-freiburg.de/~cziegler/BX/ , accessed date: 20/12/2020

[9] G. Adomavicius and A. Tuzhilin, “Towards the Next Generation of Recommender Systems : A Survey of the State-of-the-Art and Possible Extensions,” pp. 1–43.

[10]. M. Tkalcic, A. Odic, A.Kosir, and J.Tasic, “Affective labelling in a content-based recommender system for images,” *IEEE Trans.Multimedia*, vol. 15, no. 2, pp. 391–400, Feb. 2013

[11]. Ricci, F., Nguyen, Q.N.: Acquiring and revising preferences in a critique-based mobile recommender system. IEEE Intelligent Systems 22(3), 22–29 (2007).

[12] P. Resnick *et al.*, “GroupLens: an open architecture for collaborative filtering of a netbook,” in *Proc. the 1994 ACM Conference on Computer Supported Cooperative Work*, ACM, 1994

[13]. Z. Yang, G.-A. Levow, and H. Meng, “Predicting user satisfaction in spoken dialogue system evaluation with collaborative filtering,” *IEEE J. Sel. Topics Signal Process.*, vol. 6, no. 8, pp. 971–981, Dec. 2012.

[14] Renganathan, V., Babu, A. N., & Sarbadhikari, S. N. A Tutorial on Information Filtering Concepts and Methods for Bio-medical Searching. *Journal of Health & Medical Informatics, 04*(03), 2013.

[15] J. Bobadilla, F. Ortega, A. Hernando, and A. Guti´errez, “Recommender systems survey,” *Knowl.-Based Syst.*, vol. 46, pp. 109–132, 2013.

[16] D. M. Pennock *et al.*, “Collaborative filtering by personality diagnosis: A hybrid memory and model-based approach,” in *Proc. the Sixteenth Conference on Uncertainty in Artificial Intelligence*, Morgan Kaufmann Publishers Inc, 2000.

[17]. Kruk S & Decker S Semantic Social Collaborative Filtering with FOAFRealm. Proc. Semantic Desktop Workshop in conjunction with ISWC, Galway, Ireland 2005

[18] Gong, S.; Cheng, G. Mining user interest change for improving collaborative filtering. In Proceedings of the Second International Symposium on Intelligent Information *Technology Application*, *Shanghai*, China, 21–22 December 2008; Volume 3, pp. 24–27.

[19] Schafer, J.B.; Frankowski, D.; Herlocker, J.; Sen, S. Collaborative filtering recommender systems. In *The Adaptive Web*; Springer: Berlin/Heidelberg, Germany, 2007; pp. 291–324.

[20] Deng, F. Utility-based recommender systems using implicit utility and genetic algorithm. In Proceedings of the 2015 International Conference on Mechatronics, Electronic, Industrial and Control Engineering (MEIC-15), Shenyang, China, 1–3 April 2015; *Atlantis Press: Amsterdam, The Netherlands, 2015.*

[21] Burke, R. Integrating knowledge-based and collaborative-filtering recommender systems. In Proceedings of the *Workshop on AI and Electronic Commerce*, Orlando, FL, USA, 18–22 July 1999; pp. 69–72.

[22]. Burke, R. Hybrid web recommender systems. In: The Adaptive Web, pp. 377–408. Springer Berlin / Heidelberg (2007)

[23] N. Zheng, L. Qiudan, L. Shengcai, Z. Leiming, "Which photo groups should I choose? A comparative study of recommendation algorithms in Flickr," *Journal of Information Science*, vol. 36 no. 6, pp. 732–750, 2010.

[24] E. Brynjolfsson, Y.J. Hu, M.D. Smith, "Consumer surplus in the digital economy: estimating the value of increased product variety at online booksellers," *Forthcoming in Management Science*, vol. 49, no. 11, pp. 1580–1596,2003.

[25] Tranos Zuva, Sunday O. Ojo, Seleman M. Ngwira, and Keneilwe Zuva, "A Survey of Recommender Systems Techniques, Challenges and Evaluation Metrics," *International Journal of Emerging Technology and Advanced Engineering*, vol. 2, no. 11, 2012.

[26] Rong Hu and Pearl Pu., "Using Personality Information in Collaborative Filtering for New Users," *in preceding semantic scholar*, 2015.

[27] S. K. Tiwari and H. Potter, "An Approach for Recommender System by Combining Collaborative Filtering with User Demographics and Items Genres," *International Journal of Computer Applications*, vol. 128, no. 13, pp. 16–24, 2015.

[28] L. Safoury and A. Salah, "Exploiting User Demographic Attributes for Solving Cold-Start Problem in Recommender System," *Lecture Notes on Software Engineering*, vol. 1, no. 3, pp. 1–5, 2013.

[29] J. Bobadilla, F. Ortega, A. Hernando, and A. Gutiérrez, "Recommender systems survey," *Knowledge Based System*, vol. 46, pp. 109–132, 2013.

[30] J. Lu, D. Wu, M. Mao, W. Wang, and G. Zhang, "Recommender System Application Developments: A Survey," *Decision Support Systems*, Vol. 74, pp. 1– 38, 2015.

[31] S. Solanki and S. Batra," Recommender System using Collaborative Filtering and Demographic Characteristics of Users," *International Journal on Recent and Innovation Trends in Computing and Communication*, Vol.3, No.7, 2015.

[33] Anwar, K., Siddiqui, J. and Sohail, S. S 'Machine learning-based book recommender system: A survey and new perspectives, International Journal of Intelligent Information and Database Systems, 13(2–4), pp. 231–248. DOI: 10.1504/IJIIDS.2020.109457, 2020

[34] Ali, Z., Khusro, S. and Ullah, I. 'A hybrid book recommender system based on Table of Contents (ToC) and association rule mining, ACM International Conference Proceeding Series, 09-11-May-2016, pp. 68–74. DOI: 10.1145/2908446.2908481., 2020

[35] Anwar, K., Siddiqui, J. and Sohail, S. S 'Machine learning-based book recommender system: A survey and new perspectives, International Journal of Intelligent Information and Database Systems, 13(2–4), pp. 231–248. DOI: 10.1504/IJIIDS.2020.109457, 2020

[36] Jomsri, P., 'FUCL mining technique for book recommender system in library service', Procedia Manufacturing. Elsevier B.V., 22, pp. 550–557. DOI: 10.1016/j.promfg.2018.03.081, 2018

[37] Ricci, F., Rokach, L. and Shapira, B. Recommender Systems Handbook, Recommender Systems Handbook. DOI: 10.1007/978-0-387-85820-3., 2011

[38] Ali, Z., Khusro, S. and Ullah, I. 'A hybrid book recommender system based on Table of Contents (ToC) and association rule mining, ACM International Conference Proceeding Series, 09-11-May-2016, pp. 68–74. DOI: 10.1145/2908446.2908481., 2016

[39] M. Surendra and P. Babu, “An Implementation of the User-based Collaborative Filtering Algorithm,” *International Journal of Computer Science and Information Technologies*, vol. 2, no. 3, pp. 1283–1286, 2011.

[40] Isinkaye, F. O., Folajimi, Y. O., & Ojokoh, B. A. Recommendation systems : Principles, methods and evaluation. *Egyptian Informatics Journal*, *2015*(16), 261–273.

[41] F. O. Isinkaye, Y. O. Folajimi, B. A. Ojokoh, “Recommendation systems: Principles, Methods and Evaluation,” *Egyptian Informatics Journal*, vol. 16, pp. 261-273, 2015.

[42] A. Darvishy, H. Ibrahim, A. Mustapha, and F. Sidi, “New Attributes for Neighbourhood-based Collaborative Filtering in News Recommendation,” *Journal of Emerging Technologies in Web Intelligence*, vol. 7, no. 1, pp. 13–19, 2015.

[43] Urszula Kużelewska “Clustering Algorithms in Hybrid Recommender System on MovieLens Data,” *Studies in Logic, Grammar and Rhetoric*, vol. 37, no. 50, pp. 125–139, 2014.

[44] Tranos Zuva, Sunday O. Ojo, Seleman M. Ngwira, and Keneilwe Zuva, “A Survey of Recommender Systems Techniques, Challenges and Evaluation Metrics,” *International Journal of Emerging Technology and Advanced Engineering*, vol. 2, no. 11, 2012.

[45] Le Hoang Son, “Dealing with the new user cold-start problem in recommender systems: A comparative review,” *in press. Information Systems*, 2014.

[46] P. Melville and V. Sindhwani, “Recommender Systems,” i*n Encyclopedia of Machine Learning, Springer*, pp. 829-838, 2010.

[47] L. Sharma and A. Gera, “A Survey of Recommendation System: Research,” *International Journal of Engineering Trends and Technology*, vol. 4, pp. 1989–1992, 2013.

[48] Renganathan, V., Babu, A. N., & Sarbadhikari, S. N. A Tutorial on Information Filtering Concepts and Methods for Bio-medical Searching. *Journal of Health & Medical Informatics*, *04*(03), 2013

[49] Uko, E., O., B. and O., P. ‘An Improved Online Book Recommender System using Collaborative Filtering Algorithm’, International Journal of Computer Applications, 179(46), pp. 41–48. DOI: 10.5120/ijca2018917193,2018

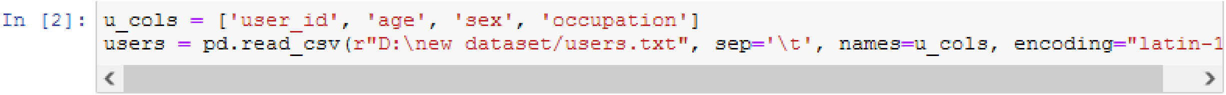
[50] P. M. Rosa, J. J. P. C. Rodrigues, and F. Basso, “A weight-aware recommendation algorithm for mobile multimedia systems,” *Mobile Inf. Syst.*, vol. 9, no. 2, pp. 139–155, 2013.

[51] J. Yau and M. Joy, “A context-aware and adaptive learning schedule framework for supporting learners’ daily routines,” in *Proc. 2nd IEEE Int. Conf. Syst.*, Martinique, France, Apr. 2007, pp. 31–37.

[52] K. Verbert, N. Manouselis, X. Ochoa, M. Wolpers, H. Drachsler, I. Bosnic, and E. Duval, “Context-aware recommender systems for learning: A survey and future challenges,” *IEEE Trans. Learning Technol.*, vol. 5, no. 4, pp. 318–335, Oct.–Dec. 2012.

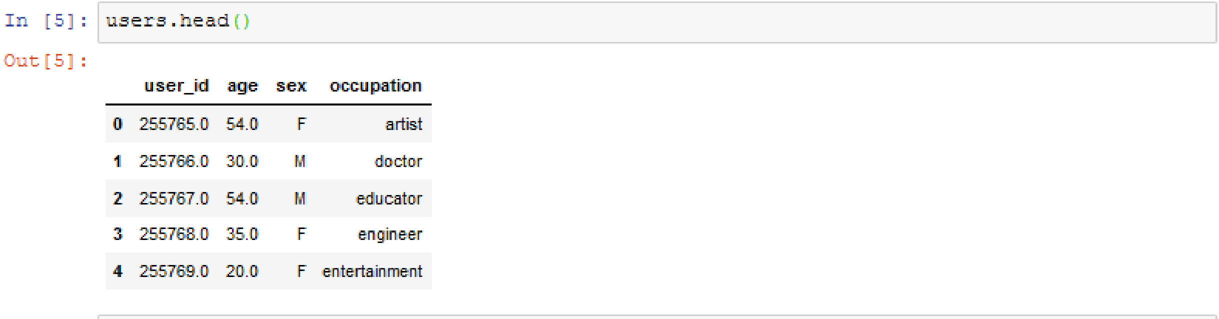
[53] M. Sunitha and T. Adilakshmi, “Recommender Systems to Address New User Cold-Start Problem with User Side Information,” *IOSR Journal of Computer Engineering (IOSR-JCE)*, vol. 18, no. 2, pp. 17–23, 2016.

Appendix I, Sample python code the proposed model.



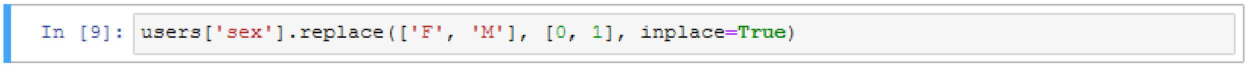
```
In [2]: u_cols = ['user_id', 'age', 'sex', 'occupation']
        users = pd.read_csv(r"D:\new dataset/users.txt", sep='\t', names=u_cols, encoding="latin-1
```

Sample inserted dataset

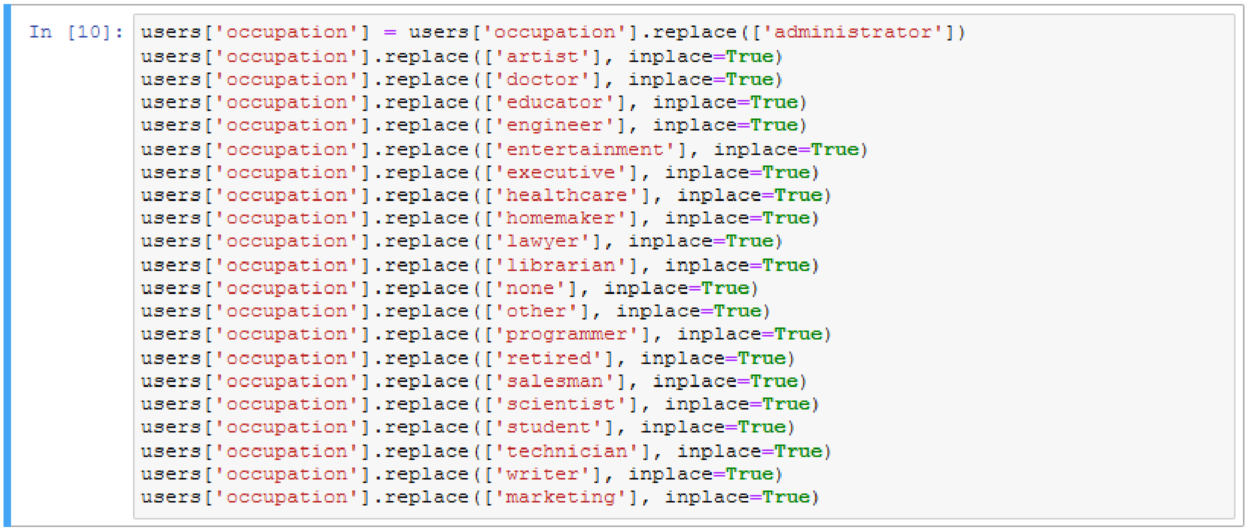


```
In [5]: users.head()
Out[5]:
```

	user_id	age	sex	occupation
0	255765.0	54.0	F	artist
1	255766.0	30.0	M	doctor
2	255767.0	54.0	M	educator
3	255768.0	35.0	F	engineer
4	255769.0	20.0	F	entertainment



```
In [9]: users['sex'].replace(['F', 'M'], [0, 1], inplace=True)
```



```
In [10]: users['occupation'] = users['occupation'].replace(['administrator'])
         users['occupation'].replace(['artist'], inplace=True)
         users['occupation'].replace(['doctor'], inplace=True)
         users['occupation'].replace(['educator'], inplace=True)
         users['occupation'].replace(['engineer'], inplace=True)
         users['occupation'].replace(['entertainment'], inplace=True)
         users['occupation'].replace(['executive'], inplace=True)
         users['occupation'].replace(['healthcare'], inplace=True)
         users['occupation'].replace(['homemaker'], inplace=True)
         users['occupation'].replace(['lawyer'], inplace=True)
         users['occupation'].replace(['librarian'], inplace=True)
         users['occupation'].replace(['none'], inplace=True)
         users['occupation'].replace(['other'], inplace=True)
         users['occupation'].replace(['programmer'], inplace=True)
         users['occupation'].replace(['retired'], inplace=True)
         users['occupation'].replace(['salesman'], inplace=True)
         users['occupation'].replace(['scientist'], inplace=True)
         users['occupation'].replace(['student'], inplace=True)
         users['occupation'].replace(['technician'], inplace=True)
         users['occupation'].replace(['writer'], inplace=True)
         users['occupation'].replace(['marketing'], inplace=True)
```

Precision, recall and F1 score result

Appendix III Experimentation value based on each user similarity

User_id	Precision	Recall	F1 score
256925	0.563332	0.398733	0.573683
269855	0.621853	0.414014	0.490567
253685	0.697038	0.57001	0.48655
268522	0.551482	0.396948	0.49964
258647	0.641113	0.420381	0.490112
289654	0.71961	0.527674	0.409364
658397	0.720742	0.40401	0.641875
258469	1	0.505264	0.654099
248932	0.781231	0.424241	0.49111
258764	0.680756	0.429294	0.599555
256983	0.741482	0.49898	0.591
256893	0.655147	0.543039	0.506317
275862	0.583123	0.398278	0.49191
255486	0.561321	0.488536	0.400376
258665	0.60717	0.461384	0.50011
Average	**0.675856**	**0.45805**	**0.5217436**

Table 5.2: Experimentation values for cluster based user similarity

Cluster Number	**Precision**	**Recall**	**F1 score**
1	0.89986	0.409639	0.571429
6	0.96911	0.421053	0.585915
10	0.98988	0.408	0.569832
15	0.891304	0.366071	0.518987
21	0.92799	0.407	0.559832
Average	**0.9355198**	**0.4023526**	**0.56119**