Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 07, No. 02, June 2021.

ENHANCING COLLABORATIVE FILTERING RECOMMENDATION USING REVIEW TEXT CLUSTERING

Ayman S. Ghabayen[1] and Basem H. Ahmed[2]

(Received: 6-Jan.-2021, Revised: 5-Mar.-2021, Accepted: 27-Mar.-2021)

ABSTRACT

The enormous rapid growth of the online world and universal computing brought a wide range of choices for Internet users to obtain information of interest. However, the huge amount of new information released every day in "big data" is greater than the human information processing capacity. As a result, it becomes harder and harder for users to obtain the required information quickly and they are also facing the problem of information overload. Collaborative Filtering (CF) systems play an important role in overcoming the information overload phenomenon by providing users with relevant information based on their preferences. CF is one of the best recommendation approaches that automate the process of the "word-of-mouth" paradigm. The most critical tasks in CF are finding similar users with similar preferences and then predicting user ratings to provide a personalized list of ranked items to the users. Previous studies have almost exclusively focused on these tasks separately to enhance the quality of recommendation. Nevertheless, we argue that these two tasks are not completely independent, but are part of an incorporated process. The purpose of this study is to propose a recommendation method that bridge the gap between the tasks of rating prediction and ranking to better grasp the best similar users to the target user by combing the advantage potential information of users review text clustering and user numerical ratings to enhance the CF recommendation methods proposed in the literature. The experimental results on three different datasets from Amazon show a considerable improvement over the baseline CF approaches in terms of recall, precision and F1-measure.

KEYWORDS

Recommender systems, Collaborative filtering, Review text, Clustering, Top-N recommendation.

1. INTRODUCTION

There has been significant growth in the digital world across the Internet in recent years. With a tremendous amount of information, it has become extremely difficult to decide with the wide-ranging of alternatives and suggestions provided to us every day. Recommender systems (RSs) can address the information overload issue, by suggesting users with recommendations of items, such as websites, movies, books, songs and music based on their individual historical preferences [1]-[2]. In e-commerce, many commercial websites, such as Amazon.com, eBay, Netflix, Yelp, last.FM, YouTube, etc. provide recommendation services. Also, in social media sites, recommender systems can help users annotate items with tags using tag recommendation, thus impeding more effective retrieval and classification in tagging systems [3].

This recommendation service could be a strategy to improve the relationship between commercial websites and their customers. Therefore, it is intended that a high-quality personalized recommendation service can ensure customers' satisfaction and loyalty [4]. CF approach is considered probably one of the most commonly applied and successful technology in RS [2], [5]. CF assumes that users who chose item A will be interested in item B if other users who chose item A were also interested in item B. CF matches the target user choices against other users to identify a group of 'like-minded', also known as 'nearest neighbour', users. This is typically done using metrics, such as cosine similarity or Pearson's correlation coefficient. Once the group of 'like-minded' users is identified, those items, which gain a high rate or are selected by the group top-N preferable items that a target user has not accessed, are then recommended [6].

Ordinary CF approaches depend on the commonality among users. Similar users or items are realized

1. A. S. Ghabayen is with the Department of Computer Science, University College of Science and Technology, KhanYounis, Palestine. E-mail: a.ghabayen@cst.ps
2. B. H. Ahmed is with the Department of Computer Science, Al-Aqsa University, Gaza, Palestine. Email: basem@alaqsa.edu.ps

by computing the similarities of common users with the active user in the rating items [5]. Generally, the CF recommender system performs well once there are sufficient user preferences. However, it suffers from certain limitations related to data sparsity and cold-start problems [7].

Data sparsity is considered one of the critical issues in collaborative filtering approaches [7]. In practice, many commercial recommender systems are based on a large dataset, where the number of items is always bigger compared to the number of users. Furthermore, most active users usually provide a rating for a rather restricted number of items. As a result, the user-item matrix used for collaborative filtering approaches could be extremely sparse, which makes it a challenge to make a useful recommendation. Moreover, data sparsity issue occurs in several situations and is specifically evident in a situation where a new user has just entered the system or a new item has just been added to the system, which is commonly known as ‘cold start’ problem [5], [7].

Typically, considering only the user rating data on items fails to completely indicate users’ similarity for two reasons 1) ratings alone does not demonstrate the reason overdue to a user’s rating and 2) users may rate items equally in the same way; however, their ratings may be based on different perspectives or item features.

To deal with the aforementioned problems, numerous approaches have been proposed by representing users and items with external knowledge resources, including user tags [5], [8], item contextual data [9] and use social data [10].

Nowadays, plenty of users often tend to provide their opinions on the Internet utilizing text. These reviews have the potential to provide a system with more details and efficient user preferences [11]. Putting it simply, user text reviews could be exploited, in combination with numerical ratings, to improve the word-of-mouth recommendation process.

This study aims to propose a recommendation method to better grasp the best alike users to the target user by combing the advantage potential information of review text clustering and user numerical ratings to enhance the CF recommendation methods proposed in literature works.

The remainder of this paper is structured as follows: Section 2 presents previous work related to CF. Section 3 then presents the proposed methods, details about the different steps which are performed, data pre-processing, standard collaborative filtering, similarity weighting, rating prediction, review clustering, item ranking and recommendation. Section 4 presents the dataset used, methodology and metrics used to examine the proposed approach and a variety of existing most common related CF recommendation algorithms to compare with. Section 5 presents the results and discussion. Lastly, Section 6 gives the overall conclusions of the work presented in this paper, as well as suggestions for future research to be performed in this field.

2. Related Work

This section presents prior work related to recommender systems. The first part presents related work utilizing clustering algorithms in recommender systems and the second part presents related work exploiting user reviews.

2.1 Clustering-based Recommender Systems

Clustering is considered as an unsupervised method of grouping content based on some obvious features, such as words or word phrases in a set of documents. In simple words, clustering is the process of grouping patterns or entities into restricted classes of similar objects. In this case, a large volume of data is classified into similar groups (related instances into clusters) [12]. Clustering has been widely utilized in a wide range of disciplines, such as image segmentation [13], information retrieval and filtering [13], text mining [14] and many other real-world applications [15].

In CF, the review clustering process works either by identifying users into groups with similar item reviews or items into groups that have the same users' preferences. Thereby, when a target user is recognized as similar to a given cluster, then items associated with these users within the related cluster are recommended to the target user. There have been various methods proposed to enhance RS accuracy by utilizing clustering methods [16]-[19]. Wang et al. [17] proposed a clustering-based CF for dealing with data sparsity problems. They initially clustered users according to their rating preferences into k clustering through the K-means clustering algorithm. Then, they introduced a formula to determine the missing rating in the user-item rating matrix to obtain a high-density matrix. The new calculated rating is used to determine the similarity of items and estimate the rating of the active user on items that have not been rated. Sarwar et al. [18] proposed a clustering approach that groups users into clusters upon their numerical rating behaviour. They confirmed that using rating clustering shows promising improvement in the recommendation accuracy on the traditional CF. Z. Cui et al. [19] introduced a recommendation model based on a time correlation coefficient called TCCF. The proposed model clusters similar users together based on the user's interest over time. Their model provides a higher-quality recommendation. In [16], CF and content-based filtering approaches were performed through clustering to identify similar users and items, respectively; after that, a personalized recommendation to the active user was made. The clustering procedure has been realized as a successful way to enhance the recommendation accuracy compared to the basic CF approach [16], [18]-[19]. However, literature reviews show that the majority of previous studies clustered users or items individually and identified the similarity between users and items based on numerical rating data. This inspires us to propose a new CF approach by performing review clustering to group reviews into clusters and locate users into groups of clusters based on their reviewing behaviour on items, in order to improve the recommendation quality of current traditional CF. K-means clustering algorithm is one of the common algorithms utilized with model-based CF system [20]. K-means clustering does a very good job when the clusters have a kind of spherical shape. Nevertheless, this algorithm is highly dependent on the user-defined variants; i.e., the number of clusters from the data and the selection on the initial centroid need to be initialized. Subsequently, different variants lead to inaccurate recommendation quality. Arthur and Vassilvitskii [21] proposed an enhanced k-means clustering algorithm called K-means++. K-means++ randomly chooses the initial centroids, then determines the subsequent centroid using the proportional probability to the squared distance from its closet existing centroid. According to [21], this algorithm shows an improvement in the speed and accuracy of the k-means, in addition to its ability to automatically identify the optimal number of clusters. We use the K-means++ clustering algorithm in this study.

2.2 Review-based Recommender Systems

Through the last decade, there has been intensive research in RS and various approaches have been proposed to improve the RS reliability and accuracy through exploring knowledge from other sources. Examples are: Knowledge-based Systems [22]-[23], Internet of Things (IoT) [24], Information Retrieval Systems [25], User Tags [5] and Neural Networks [26]. Besides, information from customers’ reviews can be exploited to provide accurate recommendations. Nilashi et al. [27] proposed a recommender system for e-tourism platforms. By utilizing the online reviews on social network sites, they applied supervised and unsupervised machine learning methods to analyze the customers’ online reviews besides using multi-criteria ratings in building their recommender system. The evaluation results confirmed that the use of online reviews leads to precise recommendations.

Terzi et al. [28] modified the traditional CF approach by identifying the similarities between users using their reviews on items, as an alternative to numerical ratings. More properly, two users are considered similar if both of them co-reviewed an item. The new similarity scores are then used as a weight in the rating prediction stage in CF.

Musto et al. [29] offered a multi-criteria CF method that makes use of users' reviews to produce a multi-faceted representation of users' interests. Furthermore, sentiment analysis and opinion mining frameworks were applied to extract relevant aspects and sentiment scores from users' reviews.

Macdonald and Ounis [30] applied a weak supervision process at the data pre-processing phase to combine both implicit and explicit users' feedback. This process was focusing on bridging the gap between the tasks of item rating prediction and item ranking. The proposed approach achieved raising the representation of less popular items in the recommendation list. Accordingly, the results showed a comparable accuracy in terms of rating prediction and item ranking as compared to other methods.

Margaris et al. [31] investigated a venue recommendation system for social network users by considering user reviews’ features related to the venues (e.g. service, price, atmosphere, physical distance), in addition to CF score which entails likeness of users’ tastes. Later, the recommendation process was provided based on both explicit rating scores and implicit scores estimated by handling textual review features. Other researchers [32], [33] used topic modeling to identify hidden topics from users' reviews exploiting topics based on latent Dirichlet allocation (LDA) for generating the topic distribution profile of users.

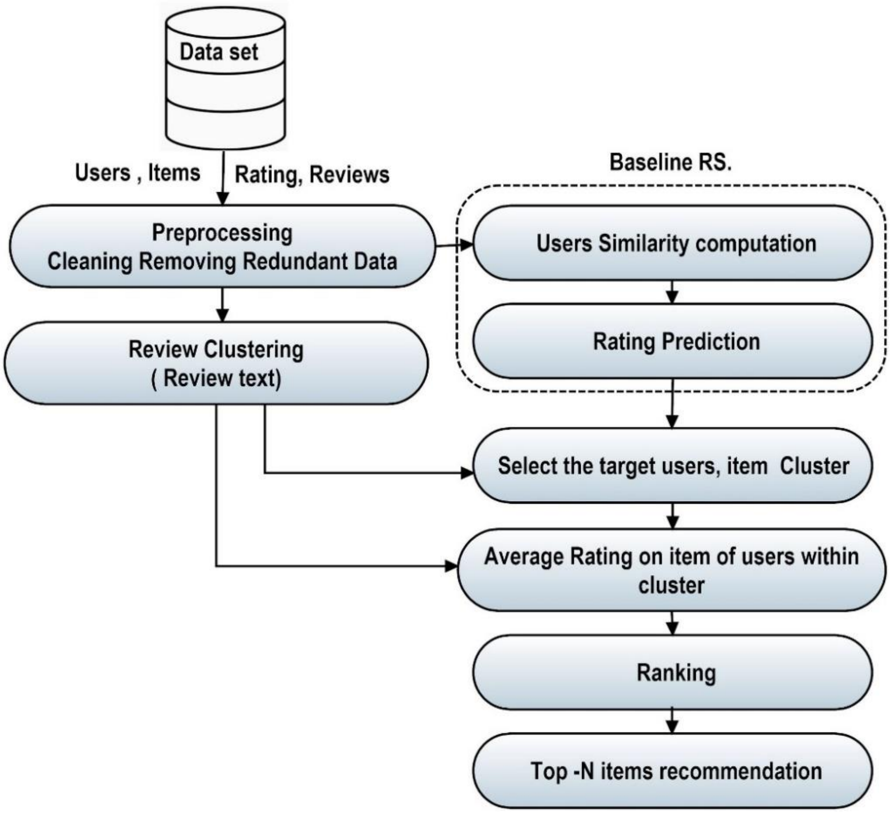

Figure 1. Proposed collaborative filtering framework.

3. Proposed Method

The main aim of this study is to propose a method to improve the recommendation accuracy by clustering users' review texts and integrating them with user ratings. Figure 1 shows the proposed collaborative filtering framework. The following subsection presents the proposed method step in more detail.

3.1 Data Pre-processing

The pre-processing step aims to refine the review text from parts that decrease the efficiency of the clustering and recommendation processes. To improve review clustering efficiency, several pre-processing steps have been made. First, redundant data and rows without review text are deleted. Next, all non-alphabetic characters, like emotion letters, smiles, finding punctuation, periods, hyphens and stop words, are eliminated from each review text. Then, the stems (roots) are identified and upper-case letters are converted into lower-case letters in the words in each review. Conversely, to avoid cold-start and sparsity problems in the recommender system, users who have fewer than 5 reviews or ratings are filtered out.

3.2 Standard Collaborative Filtering

Traditional approaches use the entire user-item database to identify the so-called “neighbourhood” of a new user or new item. Based on the neighbourhood distance or the correlation between two users or items, each neighbour receives a weight and then, the algorithm in some manner aggregates the preferences of the neighbours to produce a prediction or recommendation for the new user or (target user) [34]. Hence, when the task is to produce Top-N recommendations, these approaches tend to find the most similar (nearest neighbours) users or items. Because such an approach makes a prediction based on local similar users (neighbourhood) of the target user or similarities between items, it is commonly classified into user-based and item-based approaches [34]-[36].

The user-based collaborative filtering assumes that users who chose item A will be interested in item B if other users who chose item A are also interested in item B. On the other hand, item-based collaborative filtering looks at each item on the target user list of the chosen items and identifies other items that seem to be 'similar' to that item. The similarity of items depends on the closely matching attributes with the previously rated items by the target user.

3.3 Similarity Weighting

The most commonly used methods to calculate similarity among the two users u and v are the cosine-based and correlation-based similarity measures [37]. The similarity between user u and v is measured by calculating the cosine angle between users' corresponding rating vectors $u = (r_{n,1}, \dots, r_{n,N})$ and $v = (r_{k,1}, \dots, r_{k,N})$ defined as follows:

$$\cos(\theta) = \frac{\vec{u} \cdot \vec{v}}{\|\vec{u}\| \cdot \|\vec{v}\|} \quad (1)$$

Using Equation (1), the cosine similarity measure and the Pearson correlation coefficient between users u and v are defined respectively as follows:

$$\mathrm{Sim}\,(u,v) = \frac{\sum_{i_m \in I_{u,v}} r_{u,m} * r_{v,m}}{\sqrt{\sum_{i_m \in I_{u,v}} {r^2}_{n,m}} * \sqrt{\sum_{i_m \in I_{u,v}} {r^2}_{v,m}}} \quad (2)$$

$$\mathrm{Sim}(u,v) = \frac{\sum_{i_m \in I_{n,k}} (r_{n,m} - \bar{r}_n) * (r_{k,m} - \bar{r}_k)}{\sqrt{\sum_{i_m \in I_{n,k}} (r_{n,m} - \bar{r}_n)^2} * \sqrt{\sum_{i_m \in I_{n,k}} (r_{k,m} - \bar{r}_k)^2}} \quad (3)$$

where $I_{n,k}$ denotes the co-rated items between users u and v. In other words, it denotes items rated by both users.

3.4 Rating Prediction

After computing the similarities between the target user/active user and the other users, k-nearest neighbours to the target user are identified. Generally, the CF estimates the rating of the unseen item for the target user based upon item rating from those k-nearest neighbours. As mentioned earlier, CF generates predictions based on the entire set of those items that have been rated or chosen by the target user. More specifically, the gain of the utility function $\hat{r}(u,i)$ of item $i \in I$ for user $u \in U$ is computed as an aggregate of the ratings r_v of most similar users for user u on item i. The utility function is defined as follows:

$$\hat{r}(u,i) = \bar{r}_u + \frac{\sum_{v \in \hat{U}} \mathrm{sim}(u,v) * (r_v - \bar{r}_v)}{\sum_{v \in \hat{U}} |\mathrm{sim}(u,v)|} \quad (4)$$

where $\bar{r}_u$ denotes the average rating of user u and $\bar{r}_v$ the average rating of user v. The average rating $\bar{r}_v$ is defined as: $\bar{r}_v = \frac{1}{|I_v|} \sum_{i_m \in I_v} r_{v,m}$, where $I_{v,m} = \{ i_m \in I \,|\, r_{v,m} \neq 0 \}$.

3.5 Review Clustering

In this step, we conduct the user review clustering process for identifying clusters, each of which is composed of a group of users or items who/that possess similar reviewing preferences among each other. Hence, this process groups users or items into clusters, thus giving a new way to identify the neighbourhood similarities of users or items in the CF recommender system.

As mentioned earlier, the objective of this research is to propose a recommendation method that combines the explicit review text data with the implicit user rating data in CF recommendations. In more detail, the Top-N recommendation list is re-ranked according to the similarity between target user/item within a related cluster, through clustering users' review data using K-means++. In this situation, an appropriate decision is made with which items might be recommended or not based on like-minded users within the cluster to improve the quality of CF recommendation. In this stage, K-means++ is applied to cluster user/item reviews.

Thereby, to cluster user/item review text, first, we need to convert the text of the free-form reviews into structure data. This means to convert the text data into numerical values. This process is sometimes referred to as “vectorization”. Among the popular vectorization processes is the term frequency-inverse document frequency (TF-IDF) measure [38]-[39]. Within the context of RS, the main idea of the TF_IDF measure is to estimate how important a keyword is to an item; the more occurrence of a keyword in a document, the more important it is. However, it also considers frequent terms that appear in many items and are not very relevant. Concretely, TF-IDF works as follows. Let N be the whole set of available documents that can be recommended to the user u_n and let N_k be the number of the documents in which the term t_k appears. First, the frequency $f_{k,m}$ of each term occurring in the document $i_m \in I$ is counted. Note that if the term t_k does not appear in the text of the document i_m, then $f_{k,m} = 0$. The term frequency of each term t_k of the document i_m is computed as follows:

$$TF_{t_k m} = \frac{f_{t_k,m}}{\max f_m} \tag{5}$$

where $\max f_m$ indicates the maximum term frequency of all terms that appear in the document i_m. In the TF measure, the more occurrence of a term in a specific document, the more important it is. However, considering terms that appear frequently in many documents tends to be less useful to determine whether the documents are relevant or irrelevant. On the other side, the inverse document frequency measure IDF is used to consider the influence of a given term in the entire collection of available documents. The IDF is regarded as a measure that minimizes the weight of terms that frequently appear in most documents, such as stop-words. Formally, IDF of the term t_k is computed as follows:

$$IDF_{t_k} = \log \frac{N}{N_k} \tag{6}$$

Finally, the TF-IDF measure for the term t_k in a document i_m is defined as the combination of term-frequency and inverse document frequency [38], which is formally defined as follows:

$$TF - IDF(t_k, i_m) = TF_{t_k,m} \times IDF_{t_k} \tag{7}$$

This method can be used to obtain terms frequently occurring in users' review text. Subsequently, using the TF-IDF, we can find out exactly what terms are important in each review. This step identifies the features of each review. Hence, the classification of reviews by TF-IDF value leads to finding a group of reviews with similar subject areas according to the importance of terms [40]. This is the reason why this research utilizes the K-means++ clustering algorithm to cluster users' reviews based on review topics. The K-means++ algorithm determines a center of the cluster that comprises a group of reviews with a specific topic and then assigns a review to a cluster based on the highest cosine similarity between the TF-IDF value of the review and the center value of each cluster. Afterwards, each review is associated with the corresponding reviewers and items.

3.6 Item Ranking and Recommendation

In Standard CF, the item ranking and recommendation step comes after the rating prediction of items that have not been evaluated by the target user. The items are ranked based upon the predicted rating values and then the Top-N items with the highest values are recommended to the target user for each recommendation interaction [41]. Accordingly, the goal of the Top-N recommendation is to obtain a list of the most relevant items allocated to user preferences. Different from the standard CF, the proposed approach recommends relevant items based on the result of review clustering and the preference propensity of each user by utilizing the estimated rating and user reviewing behaviour on items. As a result, either an incentive or a penalty is applied to each item in the Top-N recommendation list. Therefore, the Top-N list will be re-ranked based on users’ reviewing preference clustering of users on items.

The proposed ranking method is demonstrated as follows:

- Assume that the standard CF decides whether to recommend a particular item (i) to the target user (u) or not. Normally, it checks whether the predicted rate $\hat{r}(u,i)$ is sufficiently large (i.e., $\hat{r}$ >3, which means that the user likes the item), then item i will be recommended to the user u.
- Under the ranking process based on review clustering, an incentive is given to item (i) to be recommended if the estimated rate of item (i) is greater than or equal to the average estimated rate of item (i) of users within the cluster related to the active user. Otherwise, item (i) will be dropped from the Top-N recommendation list.

4. EXPERIMENTS

In this section, different experiments are conducted over three real-world datasets obtained from Amazon to examine the performance of the CF recommendation accuracy after applying the users' review clustering. All the experiments were run on a machine equipped with Intel Xeon CPU family 6, model85, CPU MHz 2000.180 and 13,021GB of RAM. The programming language used is Python 3.7.

4.1 Datasets

The proposed approach was evaluated over a real-world dataset collected by Amazon.com. In this study, 3 different dataset categories are selected for the experiment. The datasets are available at https://s3. amazonaws.com/amazon-reviews-pds/tsv/index.txt. The Amazon dataset includes product review of user reviews on each product and metadata including a numerical rating scale (1-5 stars) which indicates the user's opinion. Hence, a low rating illustrates a negative opinion, while in contrast, a high rating illustrates an incredibly positive opinion. Because of the vast size of the data, it's a challenge to handle it all. Therefore, the RS is built using a dataset of 3 product categories with the largest number of reviews; namely, Books, Video DVD and Wireless. Table 1 presents the description of Amazon dataset categories.

Table 1. Description of Amazon dataset categories.

Dataset	#Users	#Items	#Reviews
Books	4,608044	2,264749	9,292094
Video DVD	2,071004	2,97525	4,622722
Wireless	5,193777	9,06086	8,110757

4.2 Methodology and Metrics

To assess the efficiency of our proposed approach, we applied the so-called back-testing strategy, which is well known in RS evaluation. The first step was to split the dataset into 5 folds. As a result, 20% was used as testing data and 80% was used as training data. The second step was dividing each user profile into 5 folds, such that 20 % of the items are being used as testing data, while the remaining items formed the training data. This step will guarantee that the recommendation process is not biased to a certain test/ training. Besides, it also guarantees that the proposed approach produces equal recommendations for all users, not only for the most active users. Afterwards, the results were averaged over the five folds. The efficiency of the proposed approach was evaluated by using well-known standard evaluation metrics; namely, Precision, Recall and F-measure; these metrics evaluate how actually an RS can produce a highly accurate prediction for relevant items as follows:

$$\text{precision} = \frac{\text{tp}}{\text{tp}+\text{fp}} \tag{8}$$

$$\text{Recall} = \frac{\text{tp}}{\text{tp}+\text{fn}} \tag{9}$$

where tp (True positive is the number of relevant items that are to be recommended and are recommended correctly, whereas fp (False positive) is the number of non-relevant items that should have not been recommended.

tn (True negative) indicates the number of non-relevant items that should not have been recommended and were not recommended to the user and fn (False negative) is the number of relevant items that are to be recommended but are not recommended correctly. We also considered the F1-measure metric which measures the accuracy of the test. The F1-measure links both recall and precision with equal weights in a single value. This metric reflects the weighted harmonic mean of precision and recall. The F1-measure is indicated by the following equation:

$$\text{F1} - \text{meaure} = 2 \times \frac{\text{Re call} \times \text{Pr ecision}}{\text{Re call} + \text{Pr ecision}} \tag{10}$$

4.3 Comparisons

To evaluate the proposed approach performance, we implemented a variety of existing most common Baseline CF recommendation algorithms to compare with. We compare with the following baselines:

KNN: We implemented standard K-Nearest Neighbourhood CF (KNN hereinafter) [42]. This model matches the target user choices against other users to identify a group of neighbourhood users. This is typically done using similarity metrics, such as cosine or Pearson's correlation coefficient. Once the group of neighbourhood users is identified, those items which gain a high rate or are selected by the group are then recommended to the target user.

SVD: This method is one of the state-of-art model-based approaches; Standard Singular Value Decomposition (SVD hereinafter) [43]. The advantage of model-based SVD is that this model not only incorporates rating information of similar users, but also leads to obtain a rating of other users who are considered to be not similar. In this case, several users get to be predictors for other user preference events without any overlap of co-rated items. The missing user ratings are prefilling with the rating data statistics. More details on the computation of SVD are given in [44]-[45]. This method is known as a baseline predictor in several works in the literature [46]-[48].

NMF: Finally, we compare the proposed approach with CF based on Non-negative matrix factorization (NMF hereinafter) [49], this model is very similar to the SVD model and is based on the idea of rating matrix manipulation. This method reduces the dimensionality of the user-item rating matrix to a low-dimensional space and then calculates similarities between users in this space, which can enhance the recommendation efficiency. It differs from the standard SVD method in investigating a non-negative update procedure based on each feature parameter concerned instead of the entire feature matrices.

5. RESULTS AND DISCUSSION

In this section, we provide the results of our experiments concerning Top-N recommendation quality of CF. We tested each method for various values of the N-recommended items. We vary the value of N (N = 5, 10, 15, 20, 25 and 30) for each user in the test set, since in the real scenario, users tend to click on items with higher ranks. Regarding the size of the K-Nearest neighbourhood values of the standard KNN recommender system, we conducted several experiments to choose the optimal value of K. The accuracy of prediction used is the root mean square error (RMSE). RMSE computes the mean value of the differences between the actual value and the predicted value of user rating. The RMSE is given by the following equation:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(r_i - \hat{r}_i)^2} \quad (11)$$

where, r_i is the actual rating, $\hat{r}_i$ is the predicted rating and n is the number of ratings.

Table 2 shows the accuracy results of RMSE *versus* the size of the K-Nearest neighbourhood. As can be seen, on the three datasets, the RMSE values decrease when the size of the neighbourhood is increased. However, the accuracy deteriorates with k values higher than 20 and there is no significant change with k values higher than this value. Therefore, we considered the value of similar neighbourhood users K to be equal to 20.

Table 2. Accuracy results (RMSE) *versus* the size of the neighbourhood.

Dataset \ K	Books	Video DVD	Wireless
5	0.975	1.417	1.422
10	0.947	1.331	1.315
15	0.934	1.245	1.244
20	0.933	1.191	1.244
25	0.934	1.191	1.244
30	0.933	1.192	1.244
35	0.933	1.192	1.244
40	0.934	1.193	1.244
45	0.933	1.192	1.244
50	0.934	1.193	1.243

The results on the three different datasets, using Recall, Precision and F1-Measure matrices, are summarized and discussed. Table 3 presents the results of the comparison between our proposed method and the baseline approaches on the Amazon Books dataset. Hence, the best performance for each metric is shown in bold as the proposed approach abbreviation (Prop hereinafter).

Table 3. Amazon Books dataset results.

		N=5	N=10	N=15	N=20	N=25	N=30
Recall	KNN	51.1	77.9	89.2	94.0	96.3	97.4
	NMF	50.8	76.4	86.9	91.3	93.3	94.3
	SVD	50.7	76.7	87.6	92.5	94.7	95.7
	Prop	**52.9**	**78.9**	**89.7**	**94.2**	**96.4**	**97.4**
Precision	KNN	93.8	93.8	93.7	93.7	93.7	93.6
	NMF	94.3	94.0	94.0	93.9	93.9	93.9
	SVD	94.2	94.0	93.9	93.8	93.7	93.7
	Prop	**98.1**	**96.0**	**94.9**	**94.4**	**94.1**	**94.0**
F1	KNN	66.2	85.1	91.4	93.8	95.0	95.5
	NMF	66.0	84.3	90.3	92.6	93.6	94.1
	SVD	65.9	84.5	90.6	93.1	94.2	94.7
	Prop	**68.7**	**86.6**	**92.2**	**94.3**	**95.2**	**95.7**

The results in Table 3 demonstrate that the proposed approach performs better than baseline approaches in terms of Recall, Precision and F1-Measure for different values of Top-N recommendation. The proposed approach achieves the best performance when N=5. In terms of Recall, an improvement from 1.8% to 2.2 % is noticed compared to the baseline approaches. On the other hand, there is an improvement from 3.8% to 4.3 % in terms of Precision. In terms of the overall performance regarding F1-measure, the proposed approach achieves an improvement from 2.5% to 2.8%. Furthermore, it is observed that the progress of improvement has an inverse relation to the value of N. For example, the progress when N = 5 is larger than those when N = 30, for the baseline approaches and the proposed approach. This situation is due to that the most relevant items related to the target user are involved in the recommendation of Top-N values. Hence, the proposed approach can accomplish higher progress at smaller N values. The results in Table 3 demonstrate that the proposed approach is completely appropriate in a real-life scenario, since users are normally attracted first to a few number of high-ranked items [50].

We conducted two more experiments on other categories of the Amazon dataset named Video DVD dataset and Amazon Wireless dataset to study the performance of the proposed approach on other datasets with variant numbers of users, items and reviews. Table 4 and Table 5 present the result on the Amazon Video DVD dataset and Amazon Wireless dataset, respectively.

Table 4. Amazon video DVD dataset results.

		N=5	N=10	N=15	N=20	N=25	N=30
Recall	KNN	81.6	89.1	91.3	92.2	92.8	93.1
	NMF	81.2	87.4	88.9	89.6	89.9	90.1
	SVD	81.7	88.2	89.7	90.5	90.8	91.1
	Prop	**84.0**	**90.4**	**92.0**	**92.6**	**93.0**	**93.2**
Precision	KNN	91.1	91.1	91.0	91.1	91.1	90.9
	NMF	91.6	91.3	91.3	91.3	91.3	91.2
	SVD	91.4	91.1	91.1	91.0	91.0	91.0
	Prop	**93.2**	**91.8**	**91.5**	**91.3**	**91.2**	**91.1**
F1	KNN	86.1	90.1	91.2	91.7	91.9	92.0
	NMF	86.1	89.3	90.1	90.4	90.6	90.7
	SVD	86.3	89.6	90.4	90.7	90.9	91.1
	Prop	**88.4**	**91.1**	**91.7**	**92.0**	**92.1**	**92.1**

From the results in Table 4 and Table 5, we can see that the proposed approach outperforms the other baseline approaches in terms of recall, precision and F1-measure. In the case of smaller values of N, as mentioned earlier, the smaller value of N indicates a larger improvement in user satisfaction.

Table 5. Amazon Wireless dataset results.

		N=5	N=10	N=15	N=20	N=25	N=30
Recall	KNN	88.1	91.3	92.4	92.7	93.0	93.0
	NMF	87.6	89.6	90.1	90.1	90.1	90.0
	SVD	88.6	90.8	91.2	91.3	91.3	91.4
	Prop	**90.6**	**92.6**	**93.2**	**93.1**	**93.2**	**93.1**
Precision	KNN	87.3	87.7	87.5	87.7	87.6	87.5
	NMF	87.8	87.9	87.8	87.9	87.8	87.8
	SVD	87.9	87.8	87.8	87.7	87.7	87.8
	Prop	**88.3**	**87.9**	**87.8**	**87.9**	**87.8**	**87.8**
F1	KNN	87.7	89.5	89.9	90.1	90.2	90.2
	NMF	87.7	88.7	88.9	89.0	88.9	88.9
	SVD	88.2	89.3	89.4	89.5	89.5	89.6
	Prop	**89.5**	**90.2**	**90.4**	**90.4**	**90.4**	**90.4**

Finally, to examine whether the results obtained are statistically significant, a significance analysis was conducted in the form of a t-test for the proposed approach and the baseline approaches in terms of the F1-measure matrices. Hence, the F1-measure conveys the performance balance between both recall and precision. Table 6 presents the t-test results. As seen, the values of sig. (2-tailed) is less than 0.05. This ends in that the proposed approach presents a significant improvement when compared to the mentioned baseline approaches.

Table 6. T-test results on F1-meausre.

			Mean	Std. Deviation	t	Sig. (2-tailed)
Amazon Books	Pair 1	KNN	87.8333	11.25996	2.584	**0.049**
		Prop	88.7833	10.38488		
	Pair 2	NMF	86.8167	10.81026	10.808	**0.0005**
		Prop	88.7833	10.38488		
	Pair 3	SVD	87.1667	11.06882	5.525	**0.003**
		Prop	88.7833	10.38488		
Amazon Video DVD	Pair 1	KNN	90.5	2.26539	2.161	0.083
		Prop	91.2333	1.43898		
	Pair 2	NMF	89.5333	1.75575	12.912	**0.0005**
		Prop	91.2333	1.43898		
	Pair 3	SVD	89.8333	1.8085	9.037	**0.0005**
		Prop	91.2333	1.43898		
Amazon Wireless	Pair 1	KNN	89.6	0.96747	2.471	**0.05**
		Prop	90.2167	0.36009		
	Pair 2	NMF	88.6833	0.4916	27.49	**0.0005**
		Prop	90.2167	0.36009		
	Pair 3	SVD	89.25	0.5244	13.521	**0.0005**
		Prop	90.2167	0.36009		

6. CONCLUSION

Nowadays, plenty of users often tend to provide their opinions on the Internet utilizing text. These heterogeneous recommending information sources beyond user rating data present opportunities and issues for traditional CF recommender systems. In this paper, we proposed a new CF approach by utilizing review text clustering and using numerical ratings. The proposed approach aims to recommend items to a target user based on the results of review clustering and the preference tendency of each user using the predicted rating and the users' reviewing behaviour on items. In such a case, the proposed approach bridges the gap between the ranking and the prediction tasks of recommender systems, in order to better grasp the best similar users to the target user, which leads to efficiently enhance the CF Top-N recommendation. The experimental results on three different datasets show a considerable improvement over the baseline CF approaches using just user explicit rating in terms of recall, precision and F1-measure.

7. FUTURE WORK

Our future work in this area will focus on studying other clustering algorithms, in addition to the complexity of the proposed approach. Another possible direction will focus on exploring and exploiting other alternative text features in reviews.

REFERENCES

[1] X. Su and T. M. Khoshgoftaar, "A Survey of Collaborative Filtering Techniques," Advances in Artificial Intelligence, vol. 2009, Article ID: 421425, DOI: 10.1155/2009/421425, 2009.

[2] J. Bobadilla, F. Ortega, A. Hernando and A. Gutiérrez, "Recommender Systems Survey," Knowledge-based Systems, vol. 46, pp. 109-132, 2013.

[3] I. AlAgha and Yousef Abu-Samra, "Tag Recommendation for Short Arabic Text by Using Latent Semantic Analysis of Wikipedia," Jordanian Journal of Computers and Information Technology (JJCIT), vol. 06, no. 02, pp. 165-181, 2020.

[4] K. Tarnowska, Z. W. Ras and L. Daniel, "Customer Loyalty Improvement," Proc. of Recommender System for Improving Customer Loyalty Book, pp. 7-11, Cham: Springer International Publishing, 2020.

[5] A. S. Ghabayen and S. M. Noah, "Using Tags for Measuring the Semantic Similarity of Users to Enhance Collaborative Filtering Recommender Systems," International Journal on Advanced Science, Engineering and Information Technology, vol. 7, no. 6, pp. 2063-2070, 2017.

[6] J. L. Herlocker, J. A. Konstan and J. Riedl, "Explaining Collaborative Filtering Recommendations," Proceedings of the 2000 ACM Conference on Computer Supported Cooperative Work, pp. 241-250, DOI: 10.1145/358916.358995, Philadelphia, Pennsylvania, USA, 2000.

[7] S. Natarajan, S. Vairavasundaram, S. Natarajan and A. H. Gandomi, "Resolving Data Sparsity and Cold Start Problem in Collaborative Filtering Recommender System Using Linked Open Data," Expert Systems with Applications, vol. 149, p. 113248, 2020.

[8] L. Al Qadi, H. El Rifai, S. Obaid and A. Elnagar, "A Scalable Shallow Learning Approach For Tagging Arabic News Articles," Jordanian Journal of Computers and Information Technology (JJCIT), vol. 06, no. 03, pp. 263-280, 2020.

[9] G. Alshammari, J. L. Jorro-Aragoneses, N. Polatidis, S. Kapetanakis, E. Pimenidis and M. Petridis, "A Switching Multi-level Method for the Long Tail Recommendation Problem," Journal of Intelligent & Fuzzy Systems, vol. 37, pp. 7189-7198, 2019.

[10] J.-H. Su, W.-Y. Chang and V. S. Tseng, "Effective Social Content-based Collaborative Filtering for Music Recommendation," Intelligent Data Analysis, vol. 21, no. S1, pp. S195-S216, 2017.

[11] H.-A. Nuria, "A Literature Review of Word of Mouth and Electronic Word of Mouth: Implications for Consumer Behavior," Frontiers in Psychology, Mini Review, vol. 8, p. 1256, DOI: 10.3389/fpsyg.2017.01256, July 2017.

[12] A. Ghosal, A. Nandy, A. K. Das, S. Goswami and M. Panday, "A Short Review on Different Clustering Techniques and Their Applications," Proc. of Emerging Technology in Modelling and Graphics, pp. 69-83, Part of the Advances in Intelligent Systems and Computing Book Series, vol. 937, Springer, 2020.

[13] T. Nicholls and J. Bright, "Understanding News Story Chains Using Information Retrieval and Network

Clustering Techniques," Communication Methods and Measures, vol. 13, no. 1, pp. 43-59, 2019.

[14] M. Allahyari, S. Pouriyeh, M. Assefi, S. Safaei, E. D. Trippe and J. B. Gutierrez, "A Brief Survey of Text Mining: Classification, Clustering and Extraction Techniques," KDD Bigdas, Halifax, Canada, arXiv:1707.02919, 2017.

[15] K. G. Dhal, A. Das, S. Ray and J. Gálvez, "Randomly Attracted Rough Firefly Algorithm for Histogram Based Fuzzy Image Clustering," Knowledge-based Systems, vol. 216, p. 106814, DOI: 10.1016/j.knosys.2021.106814, 2021.

[16] L. U, Y. Chai and J. Chen, "Improved Personalized Recommendation Based on User Attributes Clustering and Score Matrix Filling," Computer Standards & Interfaces, vol. 57, pp. 59-67, 2018.

[17] Q. Wang, W. Cao and Y. Liu, "A Novel Clustering Based Collaborative Filtering Recommendation System Algorithm," Proc. of Advanced Technologies, Embedded and Multimedia, for Human-centric Computing, pp. 673-680, Part of the Lecture Notes in Electrical Engineering Book Series, vol. 260, Springer, Dordrecht, Netherlands, 2014.

[18] B. M. Sarwar, G. Karypis, J. Konstan and J. Riedl, "Recommender Systems for Large-scale E-commerce: Scalable Neighborhood Formation Using Clustering," Proceedings of the 5th International Conference on Computer and Information Technology (ICCIT), vol. 1, pp. 291-324, 2002.

[19] Z. Cui et al., "Personalized Recommendation System Based on Collaborative Filtering for IoT Scenarios," IEEE Transactions on Services Computing, vol. 13, no. 4, pp. 685-695, 2020.

[20] T. Mohammadpour, A. M. Bidgoli, R. Enayatifar and H. H. S. Javadi, "Efficient Clustering in Collaborative Filtering Recommender System: Hybrid Method Based on Genetic Algorithm and Gravitational Emulation Local Search Algorithm," Genomics, vol. 111, no. 6, pp. 1902-1912, 2019.

[21] D. Arthur and S. Vassilvitskii, "K-Means++: The Advantages of Careful Seeding," Proceedings of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 1027-1035, New Orleans, USA, 2007.

[22] V. Vijayakumar, V. Subramaniyaswamy, R. Logesh and A. Sivapathi, "Effective Knowledge Based Recommender System for Tailored Multiple Point of Interest Recommendation," International Journal of Web Portals (IJWP), vol. 11, no. 1, pp. 1-18, 2019.

[23] M. Dong, X. Zeng, L. Koehl and J. Zhang, "An Interactive Knowledge-based Recommender System for Fashion Product Design in the Big Data Environment," Information Sciences, vol. 540, pp. 469-488, 2020.

[24] A. Hassani, P. D. Haghighi, S. Ling, P. P. Jayaraman and A. Zaslavsky, "Querying IoT Services: A Smart Carpark Recommender Use Case," Proc. of the IEEE 4th World Forum on Internet of Things (WF-IoT), pp. 619-624, Singapore, 2018.

[25] A. Bellogín and A. Said, "Information Retrieval and Recommender Systems," Proc. of Data Science in Practice, pp. 79-96, Part of the Studies in Big Data Book Series, vol. 46, Cham: Springer International Publishing, 2019.

[26] R. Ying, R. He, K. Chen, P. Eksombatchai, W. L. Hamilton and J. Leskovec, "Graph Convolutional Neural Networks for Web-scale Recommender Systems," Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 974-983, DOI: 10.1145/3219819.3219890, 2018.

[27] M. Nilashi, O. Ibrahim, E. Yadegaridehkordi, S. Samad, E. Akbari and A. Alizadeh, "Travelers Decision Making Using Online Review in Social Network Sites: A Case on TripAdvisor," Journal of Computational Science, vol. 28, pp. 168-179, 2018.

[28] M. Terzi, M. Rowe, M.-A. Ferrario and J. Whittle, "Text-based User-kNN: Measuring User Similarity Based on Text Reviews," Proc. of the International Conference on User Modeling, Adaptation and Personalization, pp. 195-206, Part of the Lecture Notes in Computer Science Book Series, vol. 8538, Cham, Springer International Publishing, 2014.

[29] C. Musto, M. d. Gemmis, G. Semeraro and P. Lops, "A Multi-criteria Recommender System Exploiting Aspect-based Sentiment Analysis of Users' Reviews," Proceedings of the 11th ACM Conference on Recommender Systems, DOI: 10.1145/3109859.3109905, Como, Italy, 2017.

[30] A. H. Jadidinejad, C. Macdonald and I. Ounis, "Unifying Explicit and Implicit Feedback for Rating Prediction and Ranking Recommendation Tasks," Proceedings of the 2019 ACM SIGIR International Conference on Theory of Information Retrieval, [Online], Available: https://doi.org/10.1145/3341981.3344225, Santa Clara, CA, USA, 2019.

[31] D. Margaris, C. Vassilakis and D. Spiliotopoulos, "Handling Uncertainty in Social Media Textual Information for Improving Venue Recommendation Formulation Quality in Social Networks," Social Network Analysis and Mining, vol. 9, no. 1, p. 64, 2019.

[32] D. Chehal, P. Gupta and P. Gulati, "Implementation and Comparison of Topic Modeling Techniques Based on User Reviews in E-commerce Recommendations," Journal of Ambient Intelligence and Humanized Computing, vol. 2020, pp. 1-6, DOI: 10.1007/s12652-020-01956-6, 2020.

[33] M. Saraswat, S. Chakraverty and A. Sharma, "Review-based Topic Distribution Profile for Recommender Systems," Proc. of Advances in Data Sciences, Security and Applications, pp. 433-443, Part of the Lecture Notes in Electrical Engineering Book Series, vol. 612, Springer, Singapore, 2020.

[34] J. B. Schafer, D. Frankowski, J. Herlocker and S. Sen, "Collaborative Filtering Recommender Systems," Proc. of the Adaptive Web, pp. 291- 324, Part of the Lecture Notes in Computer Science Book Series, vol. 4321, Springer-Verlag, 2007.

[35] K. Peffers, T. Tuunanen, M. A. Rothenberger and S. Chatterjee, "A Design Science Research Methodology for Information Systems Research," Journal of Management Information Systems, vol. 24, no. 3, pp. 45-77, 2007.

[36] D. W. Aha, D. Kibler and M. K. Albert, "Instance-based Learning Algorithms," Machine Learning, vol. 6, no. 1, pp. 37-66, 1991.

[37] D. Jannach, M. Zanker, A. Felfernig and G. Friedrich, Recommender Systems: An Introduction, Cambridge University Press, DOI: 10.1017/CBO9780511763113, 2010.

[38] G. Salton, Automatic Text Processing: The Transformation, Analysis and Retrieval of Information by Computer, Addison-Wesley Longman Publishing Co., Inc., 1989.

[39] R. Baeza-Yates and B. Ribeiro-Neto, Modern Information Retrieval, New York: ACM Press: Addison-Wesley Longman Publishing, 1999.

[40] A. M. D. E. Hassanein and M. Nour, "A Proposed Model of Selecting Features for Classifying Arabic Text," Jordanian Journal of Computers and Information Technology (JJCIT), vol. 05, no. 03, pp. 275-290, 2019.

[41] M. Srifi, A. Oussous, A. A. Lahcen and S. Mouline, "Recommender Systems Based on Collaborative Filtering Using Review Texts: A Survey," Information, vol. 11, no. 6, p. 317, 2020.

[42] M. Deshpande and G. Karypis, "Item-based Top-N Recommendation Algorithms," ACM Transactions on Information Systems, vol. 22, no. 1, pp. 143-177, 2004.

[43] B. Sarwar, G. Karypis, J. Konstan and J. Riedl, "Application of Dimensionality Reduction in Recommender System-a Case Study," Proceedings of the ACM WebKDD 2000 Web Mining for E-Commerce Workshop, DOI: 10.21236/ada439541, 2000.

[44] Y. Koren, R. Bell and C. Volinsky, "Matrix Factorization Techniques for Recommender Systems," Computer, vol. 42, no. 8, pp. 30-37, 2009.

[45] D. Bokde, S. Girase and D. Mukhopadhyay, "Matrix Factorization Model in Collaborative Filtering Algorithms: A Survey," Procedia Computer Science, vol. 49, pp. 136-146, 2015.

[46] R. Barathy and P. Chitra, "Applying Matrix Factorization in Collaborative Filtering Recommender Systems," Proc. of the IEEE 6th International Conference on Advanced Computing and Communication Systems (ICACCS), pp. 635-639, Coimbatore, India, 2020.

[47] V. X. Chen and T. Y. Tang, "Incorporating Singular Value Decomposition in User-based Collaborative Filtering Technique for a Movie Recommendation System: A Comparative Study," Proceedings of the 2019 International Conference on Pattern Recognition and Artificial Intelligence (PRAI'19), pp. 12-15, DOI: 10.1145/3357777.3357782, 2019.

[48] Z. Xian, Q. Li, G. Li and L. Li, "New Collaborative Filtering Algorithms Based on SVD++ and Differential Privacy," Mathematical Problems in Engineering, vol. 2017, pp. 136-146, 2017.

[49] X. Luo, M. Zhou, Y. Xia and Q. Zhu, "An Efficient Non-negative Matrix-factorization-based Approach to Collaborative Filtering for Recommender Systems," IEEE Transactions on Industrial Informatics, vol. 10, no. 2, pp. 1273-1284, 2014.

[50] W. Niu et al., "Context-aware Service Ranking in Wireless Sensor Networks," Journal of Network and Systems Management, vol. 22, no. 1, pp. 50-74, 2014.

ملخص البحث:

لقد جلب النّمو السريع لعالم الإنترنت والحوسبة مدى واسعاً من الخيارات لمستخدمي الإنترنت للحصول على المعلومات ذات الاهتمام. ومع ذلك، فإنّ الكمّ الهائل من المعلومات الجديدة التي تنطلق كل يوم فيما يُعرف "بالبيانات الضّخمة" هو أكبر من قدرة الإنسان على معالجة المعلومات. ونتيجةً لذلك، يصبح الأمر أصعب فأصعب بالنسبة للمستخدمين أن يحصلوا على المعلومات المطلوبة بسرعة، وهم يواجهون أيضاً مشكلة الحِمل الزّائد من المعلومات. وتلعب أنظمة الفلترة التعاونيّة دوراً مهماً في التّغلُّب على ظاهرة الحِمل الزّائد من المعلومات عن طريق تزويد المستخدمين بالمعلومات ذات العلاقة بناءً على تفضيلاتهم. وتُعدّ الفلترة التعاونيّة من بين أفضل طُرق التّوْصِيَة التي تعمل على أتمتة العمليّة المتعلقة بنموذج "الكلمة المنطوقة".

إنّ أكثر المهمات حَسْماً في الفلترة التعاونيّة هي إيجاد مستخدمين متماثلين في تفضيلاتهم، ومن ثمّ توقُّع تصنيفات المستخدمين لإنشاء قائمة شخصيّة يتمّ فيها ترتيب البنود الواردة في نصوص المستخدمين ووضعها تحت تصرُّف المستخدمين.

لقد ركزت الدراسات السابقة بشكلٍ شبه حصريّ على تلك المهمات بشكلٍ منفصل عن بعضها البعض من أجل تحسين جودة التّوصية. إلّا أننا نرى أن المُهمتيْن آنفَتَي الذّكر ليستا مستقلّتين تماماً، وإنّما هما جزء من عمليّة مُدْمَجة. من هنا، تهدف هذه الورقة الى إقتراح طريقة توصية تجسر الفجوة بين مّهمة تُوقع التّصنيفات ومّهمة التّرتيب من أجل فهمٍ أفضل للمستخدمين المماثلين للمستخدم الهَدَف، وذلك عن طريق تمشيط المعلومات المتعلقة بالنّصوص التي تصدر عن المستخدمين ووضْعها في مجموعاتٍ أو عناقيد في هيئة تصنيفات رقميّة لتحسين طُرق الفلترة التعاونيّة المقترحة في أدبيّات الموضوع. وقد أُجريت التجارب على ثلاثٍ من مجموعات البيانات التابعة لأمازون، وأسفرت النتائج عن تحسُّنٍ ملحوظ حققته الطريقة المقترحة مقارنةً بالطُرق القائمة.