DOI: http://dx.doi.org/10.26483/ijarcs.v9i2.5895

ISSN No. 0976-5697

Volume 9, No. 2, March-April 2018

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

A CLUSTERING-BASED CONTEXT-AWARE RECOMMENDER SYSTEMS THROUGH EXTRACTION OF LATENT PREFERENCES

Solomon Demissie
Ph.D Research Scholar
Department of Computer Science and Systems Engineering
Andhra University
Visakhapatnam - India

M.Shashi
Professor
Department of Computer Science and Systems Engineering
Andhra University
Visakhapatnam - India

Abstract: Recommender systems are tools that support personalization in terms of supporting navigation, sharing, and discovery of information and help users to find their desired content over the large volume of information. Recently, new research area on context-aware recommendations has emerged to provide the capability of utilizing social contents and exploit related tags and rating information and personalize the search for desired content by considering user's actual situation (contextual information). In this study, we propose an approach for clustering contextually similar information using unsupervised learning approach through K-Medoids clustering and demonstrate the extraction of latent preferences for recommending items under a given contextual cluster and study how such clusters of similar contextual information can be exploited to improve the prediction accuracy of a context-aware recommender systems. To evaluate the performance of our proposed recommendation strategy, the empirical analysis is conducted on the popular LDOS-CoMoDa dataset and we showed that our proposed approach outperforms state-of-the-art algorithms in terms of prediction accuracy of the computed recommendations.

Keywords: collaborative filtering (CF), K-Medoids algorithm, context awareness, clustering, context-aware recommendation, context-based rating prediction

I. INTRODUCTION

Context-aware recommendation has got much research attention recently because of its ability of modeling and predicting the long-term tastes and preferences of users by integrating their current situations (contextual information) into the recommendation process. In traditional recommender systems which only make recommendations based on users' preference history, the rating values given to resources by users may not always be accurate. A user, for instance, may like and give very high rating value to the product he may not interested to purchase. However, context-aware recommendation follows the notion that ratings given by users should dynamically reflect their behavior and be adapted to their current situations (context) in general [22].

Central to this paper is the aggregation and clustering of similar contextual information using clustering algorithm and the identification of latent preferences of users towards such contextual clusters, as well as users toward new and selected items in order to find relevant media content based on the detected context-cluster. As one of a data mining technique, clustering in recommender systems helps to overcome redundancy and ambiguity thereby facilitating personalization and recommendation. By clustering, the redundancy can be avoided by aggregating redundant contextual information's since the combined trend of a cluster can be more easily detected than the effect of a single context.

We previously studied the role of contextual information for item recommendation as well as for improving the prediction accuracy of context-aware recommender systems by demonstrating the extraction of latent preferences of users toward context, contexts towards items as well as users towards new and selected items for recommending items under a given context. We have also applied a Stochastic Gradient Descent (SGD) optimization function to further minimize the root mean square error (RMSE) measure of the resulted prediction capability of the latent preference models [29]. In this study, we follow up and complement our previous research work by computing clusters of similar contexts and explore latent preferences for personalizing the recommendation and prediction of target item under the given contextual clusters. As opposed to our previous work, the proposed model in this paper gives more importance to context clusters rather than individual user's context. We followed the methodology adopted in the work of Pichl et al. [20] to compute clustering of similar contextual information based on the context information extracted from the dataset. However, unlike their work, in this paper, we applied K-Medoids clustering algorithm also known as Partitioning Around Medoids algorithm (PAM) [15] to cluster the contextual information obtained from the dataset. The choice of using this algorithm comes from its suitability for clustering categorical data and its robustness as it is not affected by the presence of outliers or noise or extremes unlike clustering techniques based on K-Means [14] [7].

The remainder of this paper is structured as follows. In the next section, we focus on related works followed by introducing the reader to our proposed recommendation model that identifies the latent preferences in section 3. In Section 4, we present and discuss the results of the experiments and finally we conclude the paper in Section 5.

II. RELATED WORKS

Recently, there is a shift towards user-centric approaches to provide recommendations by incorporating the user's context information since it has been proven to be valuable information for building accurate recommender system [19][32]. There are different types of contexts that play an important role in improving personalized recommendations. According to Kaminskas and Ricci [18], such context information's distinguished into user-related (demographic information, activity, emotional state of the user), environment-related (time, weather, location), and multimedia (text or pictures the user is currently reading or looking at) contexts.

As for incorporating contextual information into a recommender system, Adomavicius et al. [9], a pioneer on context-aware recommendation system research, identifies three approaches which are contextual pre-filtering, contextual post-filtering and contextual modeling. Since we don't filter the input and output of the dataset in our work, we adopt the contextual modeling approach.

Many previous researches have shown the advantage of incorporating context information into the recommendation process. A context-aware recommendation system model that utilizes social media resources is proposed by the work presented in [4] in which social tags as well as rating information are incorporated into the model to personalize the recommendation given a particular context. One of the best research we used as a benchmark to explore latent preference models for this work as well as for our previous work [29] is the work done by Alhamid et al. [5]. The authors exploited and utilized social contents as context information and used a tensor model to leverage latent preferences associated with multiple dimensions of a user, item, and context for recommending multimedia contents.

As for improving context-aware recommender systems through clustering similar contextual information and incorporate into the recommendation process, various recommender approaches boosted with such clustered contextual information have been developed for several application domains. To mention some, Shepitsen et al. [2] proposed a personalization algorithm for recommendation in folksonomies (Collaborative tagging systems) which relies on hierarchical tag clusters. Their basic recommendation framework is independent of the clustering method, but they used a context-dependent variant of hierarchical agglomerative clustering which takes into account the user's current navigation context in cluster selection. Huang [13] explored context-aware methods to provide location recommendations matching a tourist's travel preferences and visiting context. The author specifically applied clustering methods to detect touristic locations and extract travel histories from geo-tagged photos on Flickr and then proposed a novel context similarity measure to quantify the similarity between any two contexts and develop three context-aware collaborative filtering methods, i.e., contextual pre-filtering, post-filtering and modeling. With these methods, location recommendations have been provided to the current user. Pichl et al. [20] proposed playlist aggregation pipeline to implement a novel recommender system that overcome the drawbacks of contextual pre-filtering. In their latest work, the authors were interested in how contextual clusters may be leveraged for music recommendations while ensuring that the drawbacks of the pre-filtering approach can be avoided. The authors proposed to make use of Factorization Machines (FM) [17] that directly able to incorporate the contextual clusters extracted from the names of playlists for the computation of recommendations. In this paper, we followed the technique adopted by Pichl et al. [20] in generation of clusters of similar contextual information's in the given dataset.

III. RECOMMENDATION MODEL

As described in previous sections, our proposed recommendation approach utilizes clusters of similar context information's and search to identify the latent relations between context-cluster of a selected item and the preferences of users in such clusters. These two types of associations are used to build our context-aware recommendation model for rating of items in different possible context clusters.

The context parameters we considered in our model are time, daytype, season, location, weather, social, endEmo, dominantEmo, mood, physical decision and interaction. Given clusters of similar contexts associated with a user (*u*) interacting with items (*i*), the recommendation problem is to identify a list of items i_y that will be of interest for a given user u considering a list of given cluster of similar contexts, where the rating R_u,i_y is unknown. We denote the possible list of cluster of similar contexts as CC={$cc_1, cc_2,\ldots,cc_{|CC|}$}, the set of possible items as I={$i_1, i_2,\ldots,i_{|I|}$}, and the set of users as U={$u_1, u_2,\ldots,u_{|U|}$}.

Our proposed latent preference recommendation model is presented in this section. Fig. 1 depicts the overall workflow of our recommendation model utilizing contextual clusters. In the following sections, a further description of each of the processes in the proposed recommendation framework is given.

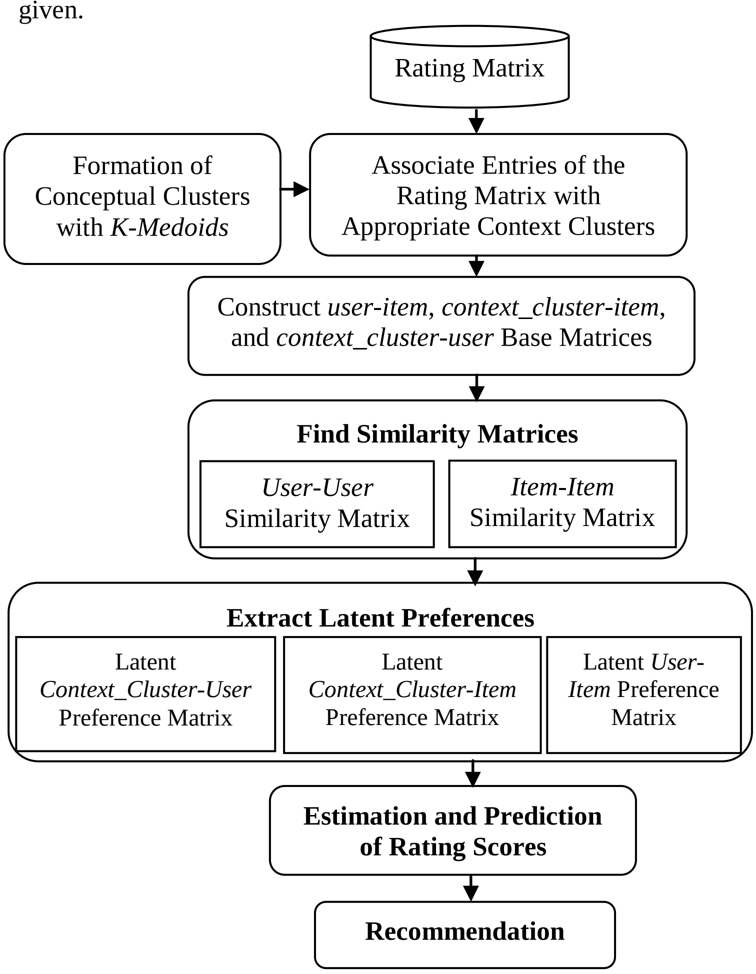

Figure 1. Pipeline for the Cluster-based Context-Aware Recommendation Model

A. *Formation of Conceptual Cluster*

Based on the context information extracted from the dataset, we compute clusters of similar contextual informations as a first step using K-Medoids clustering algorithm in which a cluster is represented by one of its points or medoid. A medoid is a cluster object having minimal distance (d) to all other objects within the cluster [16]. This is advantageous and an easy solution in terms of covering any attribute type and such medoids are proven to be resistant against outliers due to the reason that they are insensitive to peripheral cluster points [31].

K-Medoids clustering technique therefore contains two main steps. The first one is a building step in which initial k medoids are selected. Then, based on minimization of the objective function, objects (i) will be interchanged with the medoids (md_i). As shown in (1), the objective function (*OF*) is described as the sum of the distance (*d*) between all objects of the dataset to their nearest medoid. After finding the set of medoids, each object of the dataset is assigned to the nearest medoid (mdi).

$$OF = \sum d(i, md_i) \qquad (1)$$

The k-medoids clustering algorithm used a dissimilarity matrix as input for its process and this matrix measures the distance between each observation. Hence, selection of distance algorithm is the vital part of the clustering process since it directly affects the clustering results and it is dependent on the type of data to be clustered. In LDOSCoMoDa[1] dataset, all the contextual variables we used for our recommendation model is of categorical type. Accordingly, the Gower's General Similarity Coefficient [10] is applied for our experimental analysis since it is suitable and useful for measuring proximity of both numerical as well as categorical data types. As shown in (2), the Gower's General Similarity Coefficient GS_{ij} compares two cases *i* and *j*.

$$GS_{ij} = \frac{\sum_k W_{ijk} GS_{ijk}}{\sum_k W_{ijk}} \qquad (2)$$

where GS_{ijk} denotes the contribution provided by the k^{th} variable and is usually 1 or 0 depending upon whether or not the comparison is valid for the k^{th} variable. The effect of the denominator is to divide the sum of the similarity scores by the number of variables; or by the sum of their weights if variable weights have been specified.

For evaluating the quality of the discovered clusters, the silhouette index is adopted in this paper. According to [6], the silhouette refers to a method of interpretation and validation with respect to consistency within clusters of data. Its value is a measure of how similar an object is to its own cluster (cohesion) compared to other clusters (separation). As shown in (3), the silhouette is based on the mean score for every point in the data set. Each point's individual score is based on the difference between the average distance of that point to other points in its cluster and the minimum average distance between that point and the other points of other clusters. This difference is then divided by a normalization term, which is the average with the larger value,

$$Sil = \frac{1}{N} \sum_{i=0}^{N} s_{x_i} \qquad (3)$$

where, N is the number of points in the data set, and

$$s_{x_i} = \frac{(b_{q,i} - a_{p,i})}{\max\{a_{p,i}, b_{p,i}\}} \qquad (4)$$

[1] *LDOS-CoMoDa data set, http://www.ldos.si/comoda.html*

If x_i is a point in cluster *p*, then $b_{q,i} = min\ (d_{q,i})$ where $b_{q,i}$ is the average distance between point x_i and every point of cluster *q*. On the other hand, $a_{p,i}$ is the average distance between point x_i and every other point of cluster *p*. The score range is between −1 and 1, indicating that as clustering improves, then the score will approach a value of 1 [6].

After plotting the silhouette score averaged over all the contextual variables against different values of *k*, the right number of clusters is estimated to be the *k* yielding the highest average silhouette score. According to [25], the higher value returned from the Silhouette index, the better the clusters are. The plot in Fig. 2 shows the average silhouette numbers for each *k* and accordingly, the optimal number of clusters to be adopted for the clustering method we applied is 24. By assuming that this assertion is valid, we apply the identified number of clusters onto the *k*-Medoids algorithm to generate meaningful clusters of similar contexts and then assemble such contextual clusters together with user, item and rating information as input for the next process.

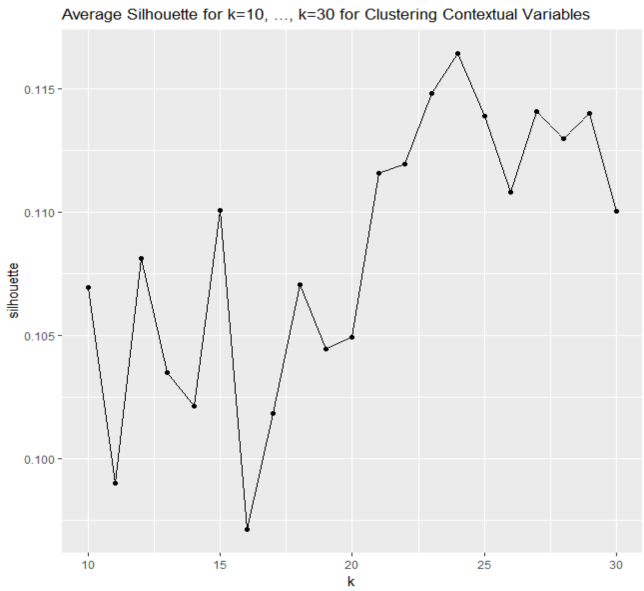

Figure 2. The graph of average silhouette value vs. k.

B. *Construct Base Matrices*

In the proposed recommendation model users' contexts for items is represented as tuples of <*user,item,context*>, which conform to a 3-dimensional matrix or a tensor. According to [8], the two main approaches for computing recommendation problems are by unfolding the 3-dimensional tensor in three bi-dimensional matrices or directly using the 3-dimensional tensor. After aggregating and clustering similar contexts, we applied the former approach and decomposed the tensor model into three bi-dimensional matrices: *user-item*, *context_cluster-item*, and *context_cluster-user* matrices.

By following the model proposed by Kim et al. [23], given a list of users *U*, a list of items *I*, and a list of contextual clusters *CC*, we can build the required matrices needed for our recommendation model. Accordingly, we construct three main matrices: *context_cluster-user* matrix $\mathbf{CCU}_{|CC|\times|U|}$ which represents the number of times a user u_y consumed items in context cluster cc_y (if a user has not consumed any items in a given context-cluster, then the CCU (cc_x,u_y) = 0), *user-item* matrix $\mathbf{UI}_{|U|\times|I|}$ that is built from the rating values assigned to items by users, and finally *context_cluster-item* matrix $\mathbf{CCI}_{|CC|\times|I|}$ which represents the frequency of items I_y selected in a particular *context_cluster* cc_y. To discover the related similarities between each individual dimension, user-user ($\mathbf{S}_{|U|\times|U|}$), and item-item ($\mathbf{T}_{|T|\times|T|}$) similarity matrices are also built as discussed in the following section.

C. Similarity Matrices

We construct two similarity matrices between users and items to discover the latent preferences in our model and accordingly leverage relevant items for a user in a particular context cluster. The similarity computation can be done by using cosine similarity measure [see (5)] which is used for measuring the similarity between objects that are represented as vectors [26]. Due to its constant superior performance for different recommendation algorithms [27][28][30], we preferred to apply cosine similarity technique in this paper.

$$sim(V_x, V_y) = \cos(V_x, V_y) = \left(\frac{V_x \bullet V_y}{\|V_x\|^2 \bullet \|V_y\|^2}\right) \quad (5)$$

To compute the user-user similarity $\mathbf{S}_{|U|\times|U|}$ as well as item-item similarity $\mathbf{T}_{|I|\times|I|}$, we first decompose matrices of the original 3-D tensor to construct the user-item matrix $\mathbf{UI}_{|U|\times|I|}$ which is built from the rating values assigned by users to items.

D. Extract Latent Preferences

By looking at the relationship between users and items for different context attributes, we can notice that there are hidden (latent) reasons why users prefer to select certain items in a given context as well as hidden causes for which an item is selected in a certain context. We can also likely notice that users who select items in particular context may also select similar items in similar contexts [22]. Accordingly, after discovering clusters of similar contextual information, our proposed model assumes that users who select items in particular context cluster may also select similar items in similar context cluster. So, our main objective is discovering the hidden (latent) features of context-clusters for both users and items based on the similarity matrices created in the previous section. To attain this objective, we analyze the contextual cluster associated with the interactions of <*user*, *item*> in the dataset. By tracing the patterns of the clustered contextual selection, we fill the gap between users and new items as well as between items and new context-clusters. The assumption in our proposed recommendation model is that there are items in I for users in U under context-cluster CC, where the user's preferences are unknown. However, three latent models can be constructed that represent the latent preferences of users toward context-clusters $\mathbf{CCTU}_{|U|\times|CC|}$, the latent preferences of items toward context-clusters $\mathbf{CCTI}_{|CC|\times|I|}$, and the latent preferences of users toward items $\mathbf{UTI}_{|U|\times|I|}$.

It is likely that in a specific context users can consume items similar to their past preferences or the preferences of similar users. Based on this notion, the context-user matrix **CCTU** can be derived from the product of the user-user similarity matrix **S** and matrix **CCU** as shown in (6). The latent preference matrix **CCTU** represents hidden context-cluster of a given user u_x and shows how a particular context-cluster was consumed by users similar to user u_x.

$$CCTU = \overline{CCU}\, S^T \quad (6)$$

where $\overline{CCU}_{|CC|\times|U|}$ is the normalized *context_cluster-user* matrix and $\mathbf{S}_{|U|\times|U|}$ is the *user-user* similarity matrix. Considering only the frequency of usage for a particular context-cluster within the users' scope might affect the recommendation accuracy by the number of users who frequently consume items in different context-clusters. Because of a small number of users who consume many items in a particular context cluster, the importance of how many users have consumed items within that context cluster would be neglected. Therefore, we normalized the frequency values in a range between 0 and 1 by the following formula:

$$ccu(cc_x, u_y) = \frac{n_{cc,u}(cc_x, u_y)}{N_{cc_x,u}} \quad (7)$$

where $n_{cc,u}(cc_x, u_y)$ is the number of occurrences of context-cluster cc_x in the list of consumed items by u_y and as (8) shows, $N(cc_x,u)$ represent the number of times the context-cluster cc_x is used by all users.

$$N_{cc_x,u} = \sqrt{\sum_{y=1}^{|U|}\left(\beta_{x,y} f_{x,y}\right)^2} \quad (8)$$
$$\beta_{x,y} = 1\; cc_x \text{ occurred in } u_y$$
$$\text{or 0 otherwise}$$

It is important to consider the effect of certain active users that consume different items in different context-clusters since the contribution of such active users in the final recommendation results is more than the less active ones. To reduce such contribution effect, we normalized the matrix that holds normalized columns for each user as shown in (6).

In the same way, the latent preferences of items toward their detected context-cluster can be predicted by taking how a particular context-cluster is behaving with the user's selection of items - in terms of items rather than users. Such hidden preference is captured by matrix $\mathbf{CCTI}_{|CC|\times|I|}$ which obtained by the product of the normalized frequency matrix of *context_cluster-item* matrix $\overline{CCI}_{|CC|\times|I|}$ and the transpose of item-item similarity matrix **T** as in (9). Column vector normalization step is utilized to normalize the frequency matrix **CCI** as we did for normalizing matrix **CCU**.

$$CCTI = \overline{CCI}\,(T)^T \quad (9)$$

The final step in exploring latent preferences is finding hidden preferences of users toward items which represented by the matrix $\mathbf{UTI}_{|U|\times|I|}$. Such latent preference can be obtained by multiplying the original normalized rating matrix **UI** with the user-user similarity matrix **S** as in (10). According to [22], the product of **UI** and **S** matrices signifies user's as well as their nearest neighbors' preferences for a given item. Normalizing the values in matrix **UI** is essential because of the reason that some users are more active in rating different items than other inactive users. This leads to more contributions in the recommendation model from the active users compared to the less active once [29]. Accordingly, we followed the same normalization step to normalize the rating matrix **UI** as we did for **CCI** and **CCU** matrices.

$$UTI = \overline{UI}^{T} (S)^{T} \qquad (10)$$

Finally, to associate the user-item relationship to each of the context-cluster, our recommendation model utilizes the two hidden preferences matrices (the CCTU and CCTI matrices). While considering user preferences, this method provides item recommendations to a given context-cluster. Accordingly, the rating values for user-item can be estimated by computing the CCTU and CCTI matrices as shown in (11).

$$Rating\_Score_{u,cc}(i) = CCTU_{cc,u} \times CCTI_{cc,i} \qquad (11)$$

where $CCTU_{cc,u}$ is the entry value of the *CC-th* row and the *U-th* column in the CCTU matrix, and $CCTI_{cc,i}$ represent the entry value of the *CC-th* row and the *i-th* column in CCTI matrix. Based on (11) we can be able to extract the latent preferences of user *u* according to detected context-cluster *cc*. This is by taking into consideration the user's previous items and the similarity of those items to the detected context-cluster. Based on this notion, user would get a recommendation of items with higher rating value. Such higher rating value which the recommended items obtained is the reflection of the likeliness of users towards those items in that particular context-cluster.

E. Example of Extracting the Latent Preferences

This section provides a descriptive example regarding the process used to explore the latent preference models. In this paper, our assumption is that all users' context affects item selections but not the rating values given to items. I.e., users' contextual information and the rating values given to items are independent to each other. If the rating given to items is affected by users' context, then an aggregate function such as average value can be applied to obtain overall rating values [5]. Hence, matrix $\mathbf{UI}_{|U|\times|I|}$ can be obtained based on our assumption.

The next one is the *context_cluster – user* matrix $\mathbf{CCU}_{|CC|\times|U|}$ in Table 1 which we can obtain it by aggregating users over their associated items for each context-cluster entry.

Table I. Example of Building the *Context_cluster – User* Matrix CCU

	u1	*u2*	*u3*	*u4*	*u5*
cc1	3		8	1	1
cc2	10	2		2	
cc3					1
cc4	5		17		2
cc5	12	2	3	1	4

As explained in section 3 sub-section D, matrix CCTU is normalized into a range between 0 and 1. Table 2 shows the *context_cluster – item* matrix $\mathbf{CCI}_{|CC|\times|I|}$ that is constructed and normalized in the same procedure as we did for matrix CCTU.

Table II. Example of Building the *Context_cluster – Item* Matrix CCI

	i1	*i2*	*i3*	*i4*	*i5*	*i6*
cc1	1		1	1		
cc2		1	1	2		
cc3					1	
cc4	1		1			
cc5	2	2	3	1		1

The next step is constructing two similarity matrices between users and items. Here, matrix $\mathbf{UI}_{|U|\times|I|}$ is used to find user-user similarity $\mathbf{S}_{|U|\times|U|}$ as well as item-item similarity matrices $\mathbf{T}_{|I|\times|I|}$ as illustrated in Table 3 and 4 below.

Table III. Example of Building the *Item – Item* Similarity Matrix T

	i1	*i2*	*i3*	*i4*	*i5*
i1	1	0.577	0.183		0.434
i2		1		0.316	
i3	0.577		1		
i4	0.316			1	

Table IV. Example of Building the *User – User* Similarity Matrix S

	u1	*u2*	*u3*	*u4*	*u5*
u1	1				
u2		1		0.912	
u3	0.80	0.912	1		
u4				1	0.540

The final step is extracting the latent preference matrices (CCTU, CCTI, and UTI). The matrix CCTU will be constructed by utilizing the normalized frequency matrix of $(\overline{CCU})$ and the user similarity matrix (**S**) based on (6) and this is presented in Table 5. To estimate the weight of each context-cluster to a user (*u*), we utilized the similarity between users by retrieving the users which select the items in each context-cluster which are similar to the given user (*u*). The first prediction step for user-item recommendations is presented in Table 5 which described that for each context-cluster, the computed values are assigned to new, never before selected users as well as to users in which the context-clusters have previously consumed to select items.

Table V. An Example that shows a Prediction of the Latent Preferences of Context-Clusters toward Users

	u1	*u2*	*u3*	*u4*	*u5*
cc1	0.184	0.021	0.285	0.365	0.206
cc2	1.105	0. 590	0.136	0. 274	0. 791
cc3	0.899	0. 137	0.279	1. 363	0. 102
cc4	0.442	0.503	0.259	0. 392	0. 141

On the other hand, the latent preference of an item to a given context-cluster (CCTI) is obtained by multiplying the normalized frequency matrix of $(\overline{CCI})$ and the item similarity matrix (**T**) based on (9) [see Table 6]. To estimate the weight of each context-cluster to an item (*i*), we utilized the similarity between items by retrieving the items selected for each context-cluster which are similar to the given item (*i*).

Table VI. An Example that shows a Prediction of the Latent Preferences of Context-Clusters toward Items

	i1	*i2*	*i3*	*i4*	*i5*	*i6*
cc1	0	0.236	0	0.375	0.455	0.255
cc2	0.118	0.317	0.421	0.417	0	0.742
cc3	0.408	0.318	0	0.649	0.501	0.577
cc4	0	0.501	0.368	0.115	0.149	0.920
cc5	0.264	0.394	0	0.547	0.410	0.310

The final latent preference model (**UTI**$_{|U|\times|I|}$), is constructed by applying the exact matrix multiplication concept between the transpose of the normalized rating matrix $\overline{UI}$ and the transpose of the user-user similarity matrix $\mathbf{S}_{|U|\times|U|}$, as shown in Table 7.

Table VII. An Example that shows a Prediction of the Latent Preferences of Context-Clusters toward Items

	i1	*i2*	*i3*	*i4*	*i5*	*i6*	*i7*
u1	1.000	0.583	0	0.512	0.236	0.322	0
u2	0	0.264	0.800	0.966	1.295	0.992	0.461
u3	0.981	0.319	0.542	0.642	1.162	0.533	0.328
u4	0	0.350	0.443	0	0.971	0.912	1.101

IV. Experimental Evaluations

In this section, we introduce the experiments conducted to evaluate the proposed approach and the baseline approaches. Before focusing on the experimental setup and the evaluation measures, we start with a description of the dataset used for the evaluation.

A. Dataset

In the domain of context-aware recommendation, among the limited number of datasets available for public research, we have used the Context Movie Dataset (LDOS-CoMoDa) for the purpose of this work. Since our focus is on the contextual information, the description of the contextual dimensions and conditions in the dataset can be described in Table 8. This dataset generally contains user interaction with the system, i.e. the rating on a 5-star scale, the basic users' information, the content information about multiple item dimensions and twelve contextual information's that describe the situation when the user consumed the Item [1].

Table VIII. List of Context Information in the LDOS-CoMoDa Data

Dimension	***Contextual Conditions***
Time	Morning, Afternoon, Evening, Night
Daytype	Working day, Weekend, Holiday
Season	Spring, Summer, Autumn, Winter
Location	Home, Public place, Friend's house
Weather	Sunny / clear, Rainy, Stormy, Snowy, Cloudy
Companion	Alone, Partner, Friends, Colleagues, Parents, Public, Family
endEmo	Sad, Happy, Scared, Surprised, Angry, Disgusted, Neutral
domEmo	Sad, Happy, Scared, Surprised, Angry, Disgusted, Neutral
Mood	Positive, Neutral, Negative
Physical	Healthy, Ill
Decision	Movie choices by themselves or users were given a movie
Interaction	First interaction with a movie, Nth interaction with a movie

B. Baseline Recommender Systems

We compare our proposed clustering-based latent preference context-aware recommendation model to two state-of-the-art methods from the recommender systems literature: CF (collaborative filtering)-based system specific to the user-based one and SVD (singular value decomposition)-based recommender systems. The incorporation of context information into CF and SVD baseline approaches will be performed by a contextual pre-filtering paradigm [12] using which the contextual information is used as a label for filtering out the ratings that do not correspond to the specified contextual information before the main recommendation method is launched on the remaining selected data. In our case, by applying the pre-filtering approach, the recommendations computation is carried out on each contextual cluster separately. That means, the recommendations will be computed on a sub-dataset of the dataset limited to a certain cluster. According to [3], one major advantage of the pre-filtering approach is that it allows deployment of any of the numerous traditional recommendation techniques and by using this approach a contextual information can essentially serves as a query (or a filter) for selecting relevant rating data.

Regarding of the collaborative filtering approach [11], the idea behind its working is recommending items the *k*-nearest neighbors or neighborhood of similar users interacted with. The similarity of taste between all users (nearest neighbors) is calculated by computing pair wise user similarities using the Jaccard Coefficient [24] of the set of items each of the two users preferred to. Thus, as clearly depicted in (12), we measure the number of commonly preferred and selected items in relation to the items both users preferred to.

$$Jaccard_{i,j} = \frac{|M_i \cap M_j|}{|M_i \cup M_j|} \tag{12}$$

The second recommender system we benchmark for comparative analysis is based on SVD (singular value decomposition) [33] which extract hidden features from the user-item rating matrix to predict user preference ratings to items. These latent (hidden) features are calculated by factorizing the rating matrix R into two lower rank matrices *U* and *V* (*R*=*UV*') which characterize the user and item factors. By applying the most successful optimization technique called stochastic gradient descent (SGD) [33], the user and item factors, *U* and *V*, are approximated by minimizing the error to the known ratings.

The above baseline recommender systems finally aim to recommend items for a given user in a given context. This will be done by modeling users by the item they desire to and supplement contextual information in which each user has selected those items. In the process of preparing the dataset to our experiment, we transform the quadruple dataset that contains <*user*, *item*, *context*, *rating*> into quadruples of <*user*, *item*, *context_cluster*, *rating*> by applying the clustering method we presented in section 3 sub-section A. Hence, each user-item pair is assigned with one of the contextual clusters we obtained in which the given user has shown preference to the given item. In this paper, we transform the task of the recommendation computation into rating prediction task by utilizing the explicit feedback given in the dataset as a rating on a 5-star scale and incorporate such information as a fourth dimension. The next section presents a detailed description of the experimental evaluation we conducted.

C. Experimental Setup

By following the procedure described in [21], we use an offline experiment and evaluate the performance of the baseline recommender systems by conducting a 5-fold cross-validation. Accordingly, we randomly partition the dataset into five folds of equal size: four folds as training data and the remaining fold as test data. We repeat the process 5 times so that every fold serves as test data once. Random selection of the data for the folds affects each fold to contain arbitrary number of relevant and irrelevant items. Those items which are selected and rated within a certain cluster are relevant items and those which a user didn't show preference to at all within a cluster are the irrelevant once. The rating prediction performance of the baseline recommender systems will be assessed by computing the predicted rating for each item and then compare such predicted rating to the actual ratings for the current user, item and cluster in the test set as described in the following section.

D. Evaluation Measures

The rating prediction task of our proposed recommendation model as well as the baseline models will be assessed by computing the two widely used error measures in recommendation systems literature: root mean square error (RMSE) and mean absolute error (MAE). The two error metrics is defined in (13) and (14) where the predicted rating, $p_{u,i}$, for user u on item i is subtracted from the actual rating, $r_{u,i}$, as contained in the test set over the total number of ratings N on the item set. We apply min-max technique to scale the predicted rating between 0 and 1 so that we can be able to compare the evaluated approaches directly.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(p_{u,i} - r_{u,i})^2}{N}} \qquad (13)$$

$$MAE = \frac{\sum_{i=1}^{n}\left|p_{u,i} - r_{u,i}\right|}{|N|} \qquad (14)$$

E. Results and Discussion

In this section, we assess and compare the predictive performance of our proposed clustering-based latent preference recommendation model and the baseline recommender models based on the experimental setup and evaluation measures described in the previous sections. By following the evaluation procedure described by Martin et al. [21], the baseline models selected for such comparative analysis with our proposed model are the user- based CF recommender system, a pre-filtering based context-aware CF recommender system [20], an SVD-based recommender system and an SVD-based context-aware recommender system with pre-filtering as well. Since the final criterion of a solution in evaluation of such baseline algorithms as well as our proposed cluster-based context-aware latent collaborative rating prediction model is based on a test set, the results of the rating prediction performance applied to all items is stated in Table 9 and in Figure 3. The result shows that our proposed model clearly outperforms all other approaches by reaching an RMSE value of 0.240 and a MAE of 0.149 respectively.

Table IX. Evaluation of Rating Prediction Performance

Recommender	***RMSE***	***MAE***
UBCF	0.295	0.187
UBCF with Contextual Pre-filtering	0.288	0.182
SVD	0.283	0.179
SVD with Contextual Pre-filtering	0.275	0.156
Our Clustered-based Context-Aware Recommendation Model	0.240	0.149

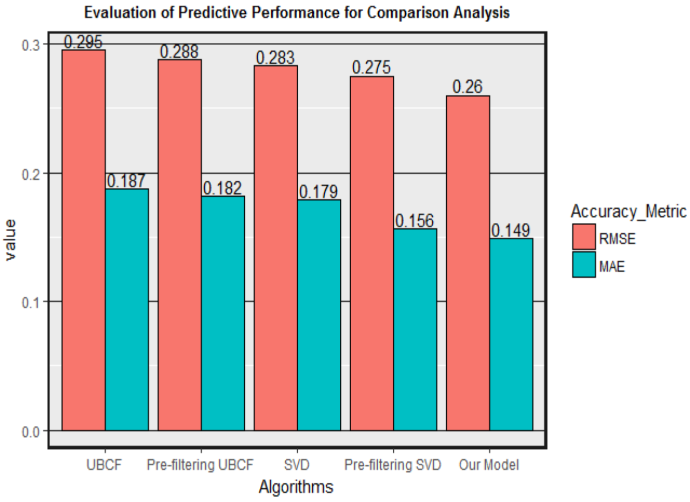

Figure 3. Comparative analysis of predictive accuracy performance.

Conversely, the contextual pre-filtering approach with respect to an SVD model-based CF algorithm is the best approach in terms of the rating prediction task by achieving an RMSE of 0.275 and MAE of 0.156 as compared with the rest of the baseline approaches and specifically with the UBCF-based contextual pre-filtering variant that reaches RMSE of 0.288 and MAE of 0.182 respectively. On the other hand, the SVD model-based CF algorithm achieved best predictive performance by scoring an RMSE value of 0.283 and MAE value of 0.179 as compared with the UBCF-based variant which scored an RMSE value of 0.295 and MAE value of 0.187. The result we achieved in terms of both RMSE and MAE value proves that the performance of the memory-based (UBCF) as well as the model-based (SVD) CF algorithms further improved by the contextual pre-filtering approaches. In general, from all the baseline approaches, the memory-based UBCF approach achieves the least prediction performance where as our proposed model showed a superior rating prediction performance and outperforms all the baseline algorithms. Furthermore, the fact that we obtain a high error rate in both RMSE and MAE value indicated the sparsity of the rating matrix we experimented and which means there are more items a user doesn't show preference to in a given cluster than a user did show preference to.

V. CONCLUSIONS

In this research work, we demonstrated a novel approach that utilizes contextual clusters to identify the latent relations between such contextual clusters of a selected item and user's preferences in such cluster and build a context-aware recommendation model for rating of items in different possible context clusters based on these two associations. We extract latent preferences based on the dimension of user, item and contextual clusters i.e., latent preference of contextual clusters towards users, latent preference of contextual clusters towards items and latent user's preference towards items. We evaluated the prediction accuracy of each of the latent preference models and showed their performance. We also propose a contextual cluster-based rating prediction model which we obtain based on the combination of the latent preferences. By applying a k-fold cross-validation evaluation technique, we evaluated the prediction accuracy performance of baseline approaches and then compare with our cluster-based rating prediction model. We obtained a result that showed the superiority of our proposed model which outperforms the baseline approaches significantly. The work we did generally show that the contribution of contextual clusters in extracting hidden preferences and to the recommendation accuracy is indeed substantial and this is a highly promising finding.

VI. REFERENCES

[1] A. Kosir, A. Odic, M. Kunaver, M. Tkalcic, and J. F. Tasic, "Database for contextual personalization," Elektrotehniski vestnik [English print ed.], vol. 78, no. 5, 2011, pp. 270–274.

[2] A. Shepitsen, J. Gemmell, B. Mobasher, and R. Burke, "Personalized Recommendation in Social Tagging Systems Using Hierarchical Clustering," Proc. ACM conference on Recommender systems (*RecSys 08*), Oct. 2008, pp. 259-266.

[3] Adomavicius, G., and Tuzhilin, A., "Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions," IEEE Transactions on Knowledge and Data Engineering, vol. 17(6), 2005, pp. 734–749.

[4] Alhamid, M.F.; Rawashdeh, M.; al Osman, H.; Hossain, M.S.; el Saddik, A., "Towards context-sensitive collaborative media recommender system," Multimed. Tools Appl. vol. 74, 2015, pp. 11399–11428.

[5] Alhamid, M.F.; Rawashdeh, M.; Dong, H.; Hossain, M.A.; el Saddik, A., "Exploring latent preferences for context-aware personalized recommendation systems," IEEE Trans. Hum.-Mach. Syst. vol. 46, 2016, pp. 615–623.

[6] Baarsch, J.; Celebi, M.E., "Investigation of internal validity measures for K-means clustering," Proc. of the International Multi-conference of Engineers and Computer Scientists, Hong Kong, China, March 2012; pp. 14–16.

[7] Bezdec, J.C., Pattern Recognition with Fuzzy Objective Function Algorithms, Plenum Press, New York, 1981.

[8] Celma, O., Music Recommendation and Discovery - The Long Tail, Long Fail, and Long Play in the Digital Music Space. Springer, 2010.

[9] G. Adomavicius and A. Tuzhilin., "Context-aware recommender systems," In Recommender Systems Handbook, springer, 2011, pp. 217–253.

[10] Gower, J., "General coefficient of similarity and some of its properties," Biometrics, vol. 27, 1971, pp. 857-874.

[11] Gediminas Adomavicius and Alexander Tuzhilin, "Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions," IEEE Trans. on Knowl. and Data Eng., vol. 17, June 2005, pp. 734–749.

[12] Gediminas Adomavicius and Alexander Tuzhilin, "Context-Aware Recommender Systems," In Recommender Systems Handbook (1st ed.), Francesco Ricci, Lior Rokach, Bracha Shapira, and Paul B. Kantor (Eds.). Springer-Verlag New York, Inc., New York, NY, USA, Chapter 7, 2010, pp. 217–253.

[13] H. Huang, "Context-aware location recommendation using geotagged photos in social media," ISPRS International Journal of Geo-Information, vol. 5(12):195, 2016.

[14] Hartigan, J. A.; Wong, M. A., "Algorithm AS 136: A K-Means Clustering Algorithm," Journal of the Royal Statistical Society, Series C vol. 28 (1): 1979, pp. 100–108.

[15] Kaufman, L. and Rousseeuw, P.J., "Clustering by means of Medoids, in Statistical Data Analysis Based on the L1 - Norm and Related Methods," edited by Y. Dodge, North-Holland, 1987, pp. 405– 416.

[16] Kaufman, L. and Rousseeuw, P., "Finding Groups in Data: An Introduction to Cluster Analysis," Wiley series in Probability and Mathematical Statistics, John Wiley and Sons Inc, 1990.

[17] M. Kaminskas, F. Ricci, "Contextual music information retrieval and recommendation: State of the art and challenges," Computer Science Review 6, 2012, pp. 89 – 119, doi:10.1016/j.cosrev.2012.04.002.

[18] Marius Kaminskas and Francesco Ricci, "Contextual music information retrieval and recommendation: State of the art and challenges," Computer Science Review 6, vol. 2, 2012, pp. 89–119.

[19] Markus Schedl, Arthur Flexer, and Julin Urbano, "The neglected user in music information retrieval research," Journal of Intelligent Information Systems 41, vol. 3, 2013, pp. 523–539.

[20] Martin Pichl, Eva Zangerle, and Gunther Specht, "Towards a Context-Aware Music Recommendation Approach: What is Hidden in the Playlist Name?" Proc. 15th IEEE International Conference on Data Mining Workshops (ICDM 2015), 2015, pp. 1360 – 1365.

[21] Martin Pichl, Eva Zangelere and G¨unther Specht, "Improving Context-Aware Music Recommender Systems: Beyond the Pre-filtering Approach," Proc. of ICMR '17, June 6–9, 2017, Bucharest, Romania.

[22] Mohammed F. Alhamid, Majdi Rawashdeh, Haiwei Dong, M. Anwar Hossain , Abdulmotaleb El Saddik, "Exploring latent preferences for context-aware personalized recommendation systems," IEEE Transactions on Human-Machine Systems, v.46 n.4, August 2016, pp.615-623.

[23] Heung-Nam Kim, Majdi Rawashdeh, Abdullah Alghamdi, and Abdulmotaleb El Saddik, "Folksonomy-based personalized search and ranking in social media services," *Information Systems*, vol. 37(1), 2012, pp. 61−76.

[24] Paul Jaccard, "Distribution of the Flora in the Alpine Zone," New Phytologist 11, vol. 2, pp. 37–50, 1912.

[25] Peter J Rousseeuw, "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis," Journal of computational and applied mathematics, vol. 20, 1987, pp.53–65.

[26] Schedl, M., Vall, A., Farrahi, K., "User Geospatial Context for Music Recommendation in Micro-blogs," Proc. ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR), Gold Coast, Australia, 2014.

[27] Shi, Y., Larson, M., & Hanjalic, A., "Mining contextual movie similarity with matrix factorization for context-aware recommendation," ACM Transactions on Intelligent Systems and Technology, vol. 4, no. 1, 2013, pp.1–19.

[28] Shin, D., Lee, J.-W., Yeon, J., & Lee, S.-G., "Context-aware recommendation by aggregating user context," Proc. of the IEEE conference on commerce and enterprise computing, 2009, pp. 423–430.

[29] Solomon D. Seifu, M.Shashi, "A context-aware recommendation system through exploring and optimizing latent preferences," International Journal of Computer Science and Network, v. 6, n. 4, August 2017, pp. 625-634.

[30] Thollot, R., "Dynamic situation monitoring and context-aware BI recommendations", Ph.D. dissertation, Ecole Centrale Paris, 2012.

[31] Velmurugan, T.; Santhanam, T., "Computational Complexity between K-Means and K-Medoids Clustering Algorithms for Normal and Uniform Distributions of Data Points," J. Comput. Sci. vol. 6, 2010, pp. 363–368.

[32] X. Liu and K. Aberer, "SoCo: A social network aided context-aware recommender system," Proc. of the 22nd international conference on World Wide Web, Rio de Janeiro, Brazil, 2013, pp.781-802.

[33] Yehuda Koren, Robert Bell, and Chris Volinsky, "Matrix Factorization Techniques for Recommender Systems," Computer 42, vol. 8, Aug. 2009, pp. 30–37.