A Clustering-based Context-Aware Recommender System through Extracting Relevant Contextual Features and Exploring Latent Preferences

Mr. Solomon Demissie Seifu[1], Prof. Shashi Mogalla[2]
Andhra University, Department of Computer Science and Systems Engineering, Visakhapatnam – 530003, India
[1]*(E-mail: soldave99@yahoo.com)*
[2]*(E-mail: smogalla2000@yahoo.com)*

***Abstract*—**Context-based information can help to increase the performance of recommender systems and enhance user satisfaction by generating an accurate recommendation. Although a complete set of contextual information is important for the recommendation process, the integration of irrelevant context features may decrease recommendation accuracy and increase computational complexity. The main aim of this work is creating contextual clusters and integrating them into the recommendation approach and extract hidden preferences for providing a recommendation of items under a given contextual cluster. Accordingly, we propose an approach that adopts three-step procedure: first, we identify relevant context attributes by obtaining their low-dimensional representation by applying a simple feature extraction method called Multiple Correspondence Analysis (MCA), a popular technique to explore the associations between levels of multiple categorical variables. Second, we run a k-means cluster analysis based on the principal components obtained from the MCA process. Six context clusters were identified and these clusters reflected the most distinct and most frequent context entities on which our latent preference-based context-aware recommender model focused. Finally, we extract 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 recommendation system. The effectiveness of our proposed method is demonstrated by the results of experiments with a real-world dataset and we show that our proposed model outperforms traditional approaches in terms of recommendation accuracy.

Keywords*—Feature extraction, Multiple Correspondence Analysis, k-means algorithm, Context-aware recommendation, Context-based rating prediction*

I. INTRODUCTION

The usability of a recommender system in practical scenarios is mainly restricted by the outcome it might bring in terms of poor accuracy. Hence in many real-life scenarios, the accuracy of recommender output is still below the practical useable threshold. Recently, the context-aware recommendation has got much research attention because of its ability of modelling and predicting the long-term tastes and preferences of users by integrating their current situations (contextual information) into the recommendation process. Such usage of context information helps to determine a personalized recommendation to a particular user in a given context and enhance the accuracy of the recommendation as well. This paper proposes to aggregate and cluster similar contextual attributes and identifies hidden 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. Hence, the effect of single context attribute is much lesser than a combined trend of context cluster in terms of facilitating personalization and accuracy of the recommendation [31].

In clustering the contextual attributes, some input context features might be irrelevant to the clustering task and these, in turn, confound the performance of the clustering process as well as the recommendation computation. In addition, such irrelevant contextual variables can act as noise and distort the computation of the pair-wise similarities among the contextual data points which is the first task ahead of the main clustering process [10]. So, one of the major challenges in this aspect is the identification of relevant context attribute and then getting the low-dimensional subspace that captures the relevant context dimensions so that the similarity graph can be defined and the underlying cluster structure can be discovered. We argue that finding the relevant low-dimensional subspace in which the structure of the contextual cluster resides would be highly beneficial for further recommendation computation. Therefore, in this paper, we introduce an approach that incorporates dimensionality reduction to find the relevant low-dimensional subspace among the contextual features based on Multiple Correspondence Analysis (MCA), a powerful factor analysis tool for categorical data which is widely used in the social and behavioural sciences [20, 21], followed by clustering those extracted low-dimensional representation of relevant context features via running k-means cluster analysis. Similar to our previous work [31], we provide new insights about the effects of clusters of similar contextual variables in the predictive performance of recommender systems and hence we give more importance to context clusters rather than individual user's context.

This paper is structured as follows. Section 2 presents an overview of relevant work about the different methods to select and extract relevant context features to apply for recommender systems. Section 3 describes our proposed recommendation model that identifies the latent preferences based on context clusters. The results of the experiments are presented and discussed in Section 4 and finally, we conclude the paper in Section 5.

II. RELATED WORKS

Many previous researches have shown the importance of selecting relevant context features to improve recommendation accuracy and decrease computational complexity in context-aware recommendation systems. To mention some, the authors in [16] conducted a context-relevance assessment to determine the influence of some pieces of contextual information on users' ratings in the tourist domain, by asking users to imagine a given situation and evaluate the influence of that contextual information. However, authors in [9] stated that such an approach is problematic, since user's rate differently in real and supposed contexts. Complementary to [16], the authors in [5] adopt two different approaches to the determination of relevant context feature: the relevancy assessment from the user survey and the relevancy detection with statistical testing on the rating data. Paired t-test technique is adopted in [11] to detect which pieces of contextual information are useful in their database and incorporate the relevant context information into their recommender system. Authors in [17] applied $\chi 2$ test to detect relevant context attributes. Rhul Gupta et al. in [27] applied a naive Bayes classifier approach to extract relevant contextual variables and adopt a Singular Value Decomposition (SVD) technique on the selected relevant contextual variables to provide the final recommendation. The authors in [2] proposed an approach that does weighing of original contextual features from the principal components to determine the most relevant contexts and then provide the recommendation based on the selected context information. Although the literature reviewed and mentioned in this section explored the importance of extracting relevant context features using various techniques, to the best of our knowledge, we did not find any literature that combines the process of extracting and clustering of relevant context features and then explores latent preferences for recommending items under a given context cluster.

III. PROPOSED RECOMMENDATION APPROACH

In this paper, our proposed recommendation approach adopted three steps: first we obtain a low-dimensional representation of the contextual variables via MCA; second, we apply k-means cluster analysis to identify a set of relatively homogenous context groups on the basis of the low-dimensional represented contextual data. Finally, we identify latent (hidden) relations between the context clusters we obtain in the second step and preferences of users in such clusters. These three types of associations are used to build our context-aware recommendation model for a rating of items in different possible context clusters.

Given context clusters coupled 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 context cluster, 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=$\{u_1, u_2, \ldots, u_{|U|}\}$.

Fig. 1 depicts the overall workflow of our recommendation model utilizing contextual clusters followed by a description of each of the processes involved in the proposed recommendation framework.

A. Extraction of Relevant Context Features using MCA

As described above, the first step in our proposed recommendation model is obtaining a low-dimensional representation of the contextual variables via Multiple Correspondence Analysis (MCA). MCA is an effective feature extraction and dimensionality reduction technique for datasets with nominal categorical data that used to detect and represent underlying structures in the data. It allows a representation between a set of dichotomous or categorical variables, in our case context attributes, in a multidimensional space of relationships that would otherwise be difficult to observe in contingency tables [32, 12]. MCA also allows the direct representation of the categorical variables as points (coordinates) in geometric space, transforming the original-

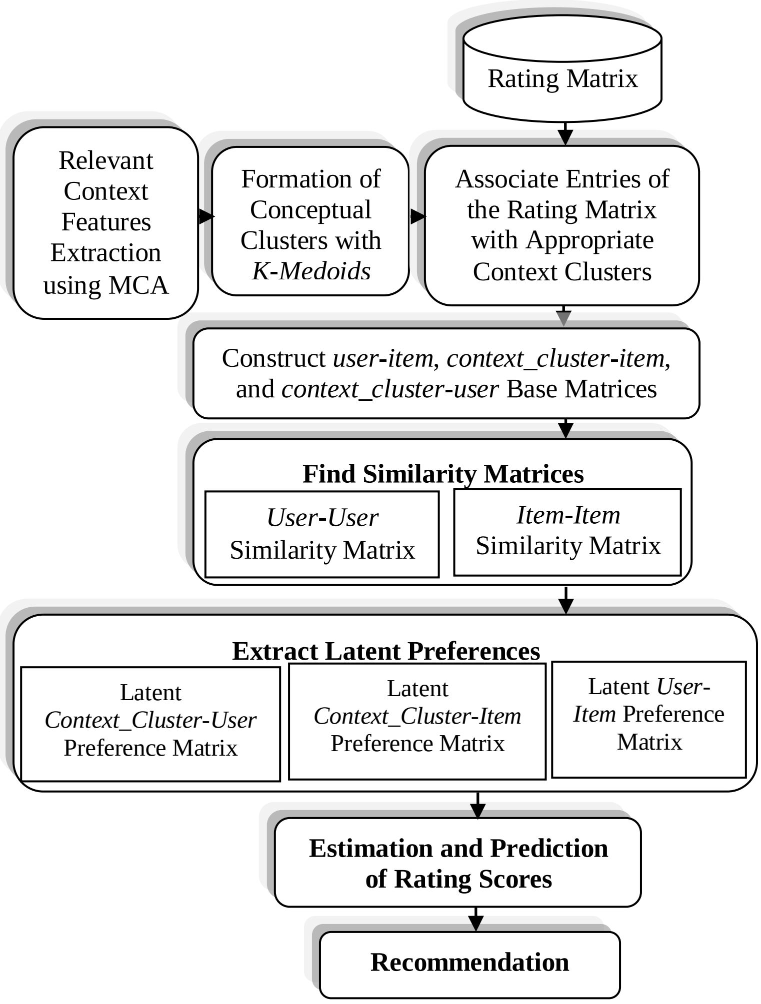


Figure 1: Pipeline for the Cluster-based Context-Aware Recommendation Model

data into continuous data. Specifically, MCA analyses a matrix of I individuals depicted by J qualitative variables by projecting the individuals into a J-dimensional space to calculate factorial axes. These projection (factorial) axes are constrained to be orthogonal in pairs in which the first axis explaining the highest possible variance and subsequent axes are having the same constraint on the residual variance. Only the factorial axes explaining a large proportion of the overall variance or the most significant factorial axes will be selected in order to reduce the number of dimensions of the initial space [18, 15, 33]. A scree plot is used to determine the percentages of inertia and the optimal number of dimensions extracted. This plot is a method to assess the most appropriate number of dimensions for interpretation and is used to present the proportions of variance explained. Inertia is an indicator of how much of the variation in the original data retained in the dimensional solution. Using the elbow rule in the scree plot, we assumed a five dimension solution to have the most accurate solution possible without including an irrelevant number of dimensions in the analysis [20].

After finding the principal dimensions or extract relevant context features via MCA, the reduced contextual dataset is applied to k-means clustering as described in the following section. The main strength of using k-means cluster analysis stems from the fact that the results are less prone to outliers in the data, the influence of the chosen distance measure, or the inclusion of inappropriate or irrelevant variables. The combined method of multiple correspondence analysis and k-means clustering is effective in analyzing extremely large datasets as no distance matrix is required [19].

B. Conceptual Cluster Formation

The second step in our proposed recommendation model is applying *K*-Means cluster analysis to identify a set of relatively homogenous context groups on the basis of the low-dimensional data obtained via MCA. Given a set of numeric objects X and an integer number *k*, the k-means algorithm searches for a partition of X into k clusters that minimizes the within groups sum of squared errors. The k-means algorithm starts by initializing the *k* cluster centres [8]. The input data points are then allocated to one of the existing clusters according to the square of the Euclidean distance from the clusters, choosing the closest. The mean (centroid) of each cluster is then computed so as to update the cluster centre [8]. This update occurs as a result of the change in the membership of each cluster. The processes of re-assigning the input vectors and the update of the cluster centres is repeated until no more change in the value of any of the cluster centres.

According to proximity criteria using the k-means algorithm with random initial centroids, the context attributes were classified in clusters from the geometric space created in MCA and clusters centres were obtained for each cluster. Determining the number of clusters in a dataset is a frequent problem in data clustering. We used Silhouette coefficients to assess the optimal number of clusters k. It is a metric that specifies how well each object lies within its chosen cluster [6]. The silhouette width of a cluster is based on the mean score for every data point in the dataset. It is the sum of each data point silhouette that contributes to this cluster. The score of silhouette ranges between -1 and 1. A silhouette of score 1 indicates a correct cluster attribution, while -1 specifies an erroneous cluster attribution and 0 stands for a data point that could either have been attributed to its present cluster or another one. The silhouette of a given data point x_i is given by the formula:

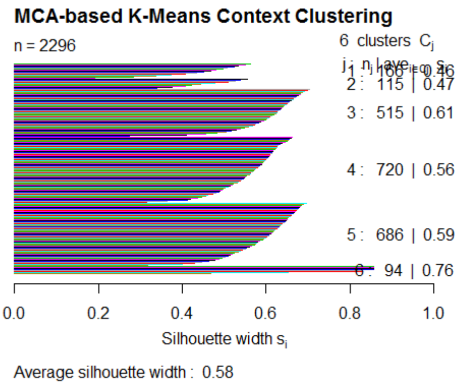
$$S_{x_i} = \frac{(b_{q,i} - a_{p,i})}{\max\{a_{p,i}, b_{p,i}\}} \quad (1)$$

If x_i is a data point in cluster *p*, then $b_{q,i} = \min(d_{q,i})$ where $d_{q,i}$ is the average distance between data point x_i and every data point of cluster *q*. On the other hand, $a_{p,i}$ is the average distance between data point x_i and every other data point of cluster *p*. After getting the silhouette value of each data point in the dataset, the score is divided by a normalization term, which is the average with the larger value, as given by Equation 2.

$$Sil = \frac{1}{N} \sum_{i=0}^{N} S_{x_i} \quad (2)$$

where, N is the number of data points in the dataset.

The optimal number of the cluster is estimated by plotting the silhouette score averaged over all the data points (context variables in our case) against different values of *k* and the *k* yielding the highest average silhouette score is selected. According to [6], the higher value returned from the Silhouette index, the better the clusters are. The first plot in Fig. 2 has the silhouette values for every low-dimensional clusters of context features and the dotted line on the second plot in Fig. 2 represents the average value of the silhouette coefficient and we can see that for k = 6 the average value of *k* is among the highest. Accordingly, the optimal number of clusters to be adopted for the clustering method we applied is 6. By assuming that this assertion is valid, we apply the identified number of clusters onto the *k*-Means algorithm to generate meaningful clusters and then assemble such contextual clusters together with the user, item and rating information as input for the next process.



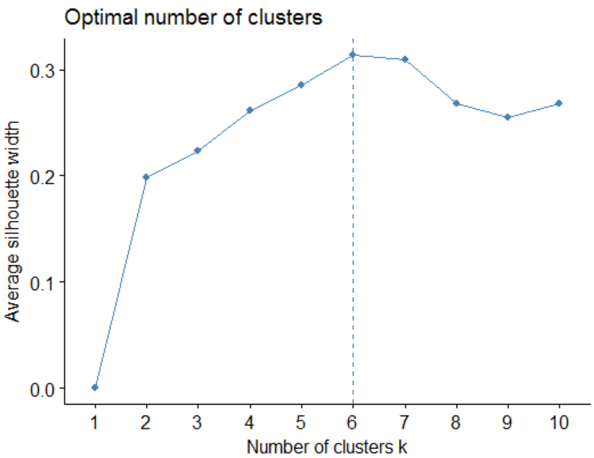


Figure 2. The graphs of average silhouette value vs. k.

C. Construct Base Matrices

Given a list of users *U*, a list of items *I*, and a list of context clusters *CC*, we construct five main matrices that are needed for our recommendation model. First, a *context_cluster-user* matrix $\mathbf{CCU}_{|CC|\times|U|}$ representing 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 $CCU(cc_x,u_y) = 0$. Second, a *user-item* matrix $\mathbf{UI}_{|U|\times|I|}$ that is built from the rating values assigned to items by users. Third, a *context_cluster-item* matrix $\mathbf{CCI}_{|CC|\times|I|}$ representing the frequency of items I_y selected in a particular *context_cluster* cc_y. Fourth, a user-user similarity matrix $\mathbf{S}_{|U|\times|U|}$ built from the user-item rating matrix $\mathbf{UI}_{|U|\times|I|}$ and finally, the item-item similarity matrix $\mathbf{T}_{|I|\times|I|}$ computed from the same matrix $\mathbf{UI}_{|U|\times|I|}$. Hence, we applied cosine similarity measure to compute the similarity of users as well as items from the user-item rating matrix [see (3)]. The Cosine similarity technique is used to determine the similarity of any two objects that are represented as vectors [28].

$$. \ sim(V_x,V_y) = \cos(V_x,V_y) = \left(\frac{V_x \bullet V_y}{\|V_x\|^2 \bullet \|V_y\|^2}\right) \qquad (3)$$

D. Extract Latent Preferences

By adopting the concept described in [24] and similar to the concept we applied in our previous work [31], our objective here is discovering hidden (latent) features of context clusters for both users and items or specifically building a model that reflects latent preferences of a given user to a context cluster as well as a given item to a context cluster by using the similarity matrices created. This is based on the notion that a relationship between users and items for different context attributes reflect latent causes for which a certain item is selected in a certain context as well as latent reasons why certain items are preferred by users in a given context. We can also notice that users who select items in particular context may also select similar items in similar contexts [24].

We analyze the context cluster associated with interactions of <*user*, *item*> and trace the patterns of the selection of the context cluster to fill the gap between users and new items as well as between items and new context-clusters. Our assumption is that there are items in *I* for users in *U* under context-cluster *CC*, where the user's preferences are unknown.

We argue that users in a specific context can consume items similar to their past preferences or similar to preferences of other similar users. Based on this notion, we can predict the latent preferences of context-cluster towards user as shown in Equation 4 that represents hidden context-cluster preference of a given user u_x and shows how a particular context-cluster was preferred by users similar to user u_x.

$$CCTU = \overline{CCU}\,(S)^T \qquad (4)$$

where $\overline{CCU}_{|CC|\times|U|}$ is the normalized *context_cluster-user* matrix and $S_{|U|\times|U|}$ is the user-user similarity matrix. Normalization of the *context_cluster-user* matrix is performed because if we consider only the frequency of usage for a particular context cluster within the user's scope, the recommendation accuracy might be affected by the number of users who frequently consume items in different context-clusters. That means, a particular context-cluster might be preferred for small number of active users who consume many items in that cluster. Due to this, the significance of item usage by other users within that context cluster would be neglected and the contribution of such active users in the final recommendation results is more than the less active ones. Therefore, to reduce such contribution effect, column vector normalization is applied to normalize the frequency values in a range between 0 and 1 as in Equation 5.

$$ccu(cc_x,u_y) = \frac{n_{cc,u}(cc_x,u_y)}{N_{cc_x,u}} \qquad (5)$$

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 Equation 5 shows, $N(cc_x,u)$ represent the number of times the context-cluster cc_x is used by all users.

$$N_{cc_x,u} = \sqrt{\Sigma_{y=1}^{|U|}\left(\beta_{x,y}\, f_{x,y}\right)^2} \qquad (6)$$

$$\beta_{x,y} = 1\, cc_x\, occurred..in\, u_y,$$

or 0 otherwise

Similarly, by using (7), we analyze how a particular context-cluster is behaving with the user's selection of items in terms of items rather than users and predict the latent preferences of items toward their detected context-cluster. Matrix $\mathbf{CCTI}_{|CC|\times|I|}$ capture such hidden preference which represents the product of the normalized frequency matrix of ($\overline{CCI}$) and the transpose of item-item similarity matrix **T.** We used column vector normalization to normalize the frequency matrix **CCI** as we did for normalizing matrix **CCU.**

$$CCTI = \overline{CCI}\,(T)^{T} \qquad (7)$$

The final step is obtaining user's latent preferences toward an item by searching items similar to their past preferences or to the preferences of similar users. Such latent preference is represented by the matrix $\mathbf{UTI}_{|U|\times|I|}$ which we build based on Equation 8.

$$UTI = \overline{UI}^{T}\,(S)^{T} \qquad (8)$$

where, matrix $\overline{UI}^{T}$ is the transposed normalized rating matrix of **UI** and **S** signifies the user-user similarity matrix**.**

Finally, the two hidden preferences matrices (the CCTU and CCTI matrices) is utilized in our proposed recommendation model to associate the user-item relationship to each of the context-cluster. Thus, in a given context-cluster, user-item rating value can be estimated by computing the product of the CCTU and CCTI matrices as shown in Equation 9.

$$Rating\_Score_{u,cc}(i) = CCTU_{cc,u} \times CCTI_{cc,i} \qquad (9)$$

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 Equation 9, the latent preferences of user *u* towards an item *i* can be extracted according to detected context-cluster *cc* and user would get the recommendation of items with higher rating value. Such higher rating value is the reflection of the likeliness of users towards those items in that particular context-cluster.

E. Example of Exploring the Latent Preferences

This section elaborates a descriptive example of the process involved to build the latent preferences models. Let us start with the graph in Fig. 3 which illustrates item ratings collected from users in different context clusters. We assume that the rating values do not change according to the context clusters; i.e., it is independent of the user's detected context cluster. This in turn means only item selections are affected by user's detected context cluster.

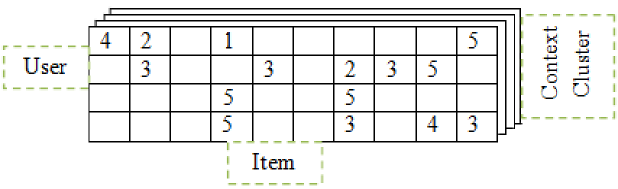


Figure 3. A Rating given by Users to Items in each Context-Cluster

Next, as Table 1 illustrates, the *context_cluster – user* matrix $\mathbf{CCU}_{|CC|\times|U|}$ can be obtained by aggregating users over their associated items for each context-cluster entry. Hence, we normalized the matrix **CCU** into a range between 0 and 1. We construct and normalize the *context_cluster – item* matrix $\mathbf{CCI}_{|CC|\times|I|}$ by following the same procedure as matrix **CCU** and this is shown in Table 2.

TABLE I. EXAMPLE OF BUILDING THE CONTEXT_CLUSTER – USER MATRIX CCU

	u1	*u2*	*u3*	*u4*	*u5*
cc1	2	1			
cc2	14	9		4	1
cc3	11	13		2	8
cc4	5				
cc5	4	5		1	14

TABLE II. EXAMPLE OF BUILDING THE CONTEXT_CLUSTER – ITEM MATRIX CCI

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

The next step is similarity computation between entries belonging to a single dimension and these are a similarity between users and items. Table 3 and 4 shows that the rating matrix $\mathbf{UI}_{|U|\times|I|}$ is utilized to find user-user similarity $\mathbf{S}_{|U|\times|U|}$ as well as item-item similarity matrices $\mathbf{T}_{|I|\times|I|}$.

The last step is constructing the matrices that reflect latent preferences (CCTU, CCTI, and UTI). The normalized frequency matrix of ($\overline{CCU}$) and the user similarity matrix (**S**) is used to create matrix CCTU and the result is presented in Table 5. Specifically, estimation of the weight of each context cluster to a user (*u*) is achieved by utilizing the similarity between users or by retrieving the users who selected the items in each context-cluster which are similar to the preference of the given user (*u*). Table 5 represent first prediction step for user-item recommendations and it describes 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 utilized to select items.

TABLE III. EXAMPLE OF BUILDING THE ITEM – ITEM SIMILARITY MATRIX T

	i1	*i2*	*i3*	*i4*	*i5*
i1	1	0.832			1
i2		1		0.316	
i3	0.832		1	0.392	
i4	0.316		0.707	1	

TABLE IV. EXAMPLE OF BUILDING THE USER – USER SIMILARITY MATRIX S

	u1	*u2*	*u3*	*u4*	*u5*
u1	1	0.40			
u2	0.40	1		0.384	
u3			1		
u4		0.38		1	0.492

TABLE V. AN EXAMPLE THAT SHOWS A PREDICTION OF THE LATENT PREFERENCES OF CONTEXT-CLUSTERS TOWARD USERS

	u1	*u2*	*u3*	*u4*	*u5*
cc1	0.060	0.061	0.285	0.058	0.084
cc2	0.407	0. 366	0.136	0. 573	0. 313
cc3	0.491	0. 622	0.279	0.299	0. 713
cc4	0.138	0.024	1	0.227	0. 059

TABLE VI. AN EXAMPLE THAT SHOWS A PREDICTION OF THE LATENT PREFERENCES OF CONTEXT-CLUSTERS TOWARD ITEMS

	i1	*i2*	*i3*	*i4*	*i5*	*i6*
cc1	0.433	0	0.037	0..067	0	0.022
cc2	0.096	0.347	0.320	0.114	0.620	0.233
cc3	0.085	0.464	0.405	0.479	0.077	0.375
cc4	0	0.050	0.006	0	0.174	0.028
cc5	0.128	0.084	0.180	0.297	0.013	0.305

TABLE VII. AN EXAMPLE THAT SHOWS A PREDICTION OF THE LATENT PREFERENCES OF USERS TOWARD ITEMS

	i1	*i2*	*i3*	*i4*	*i5*	*i6*	*i7*
u1	0	0	0.408	0	0.236	0.322	0
u2	0.641	0.516	1	0.398	1.295	0.992	0.461
u3	0	0.319	0.542	0	1.162	0.533	0.328
u4	0.273	0	0.443	0.246	0.588	0.320	1.101

By applying exact matrix multiplication concept on the normalized frequency matrix of ($\overline{CCI}$) and the item-item similarity matrix (**T**), we can obtain latent preferences of an item to a given context-cluster (CCTI) (see Table 6 above). The weight of each context-cluster to an item (*i*) can be estimated by utilizing the similarity between items by retrieving the items selected for each context-cluster which are similar to the given item (*i*).

The remaining latent preference model ($\mathbf{UTI}_{|U|\times|I|}$) is built from the normalized rating matrix $\overline{UI}$ and the user-user similarity matrix $\mathbf{S}_{|U|\times|U|}$, as shown in Table 7 above.

IV. EXPERIMENTAL EVALUATIONS

This section investigates all the experiments conducted to evaluate the item prediction accuracy of the proposed recommendation approach as well as the baseline approaches. We investigate the contribution of applying context clusters to the improvement of the recommendation performance.

A. Dataset

We utilized a dataset called LDOS-CoMoDa, a movie dataset containing user interaction with the system in terms of a rating on a 5-star scale, user's basic information, content information about multiple movie dimensions and twelve contextual attributes [1]. Based on the focus of this research, the description of all the contextual dimensions and conditions are described in Table 8.

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

Two state-of-the-art methods, the user-based collaborative filtering (UBCF) and SVD (singular value decomposition), from the recommender systems literature are investigated for comparative analysis with our proposed approach. The contextual pre-filtering paradigm [14] is used to incorporate context information into the CF-based system and the SVD one. In this paradigm, the recommendations (CF or SVD) are computed on each context cluster individually.

The first benchmark recommendation system is based on the user-based collaborative filtering approach [13] which recommends items the k-nearest neighbours or neighbourhood of similar users interacted with. To determine the nearest neighbours, the pairwise user similarities are computed using Jaccard Coefficient of the set of items each of the two users preferred to [see Equation 10].

$$Jaccard_{i,j} = \frac{|M_i \cap M_j|}{|M_i \cup M_j|} \qquad (10)$$

where M_i denotes the set of movies a user *i* has watched to.

The second benchmark approach is singular value decomposition (SVD) [35] which enables prediction of ratings by extracting latent features from the user-item matrix **UI**. The latent features which characterize types of movies in our context are computed by factorizing the rating matrix UI into two lower rank matrices *U* and *V* which represent the user and item factors. Using stochastic gradient descent (SGD) [35] optimization technique, we approximate *U* and *V* by minimizing the error to the known ratings.

The quadruple dataset that we obtain from the movie dataset with <*user*, *item*, *context*, *rating*> representation is transformed into <*user*, *item*, *context_cluster*, *rating*> to the preparation of our experiment. Each user-item pair is assigned with one context cluster within which the given user has shown preference to the given item. So, the task of the recommendation computation is transformed into a rating prediction task by incorporating each context cluster value as a fourth dimension. The next section provides detail explanation of the experimental evaluation we performed.

C. Experimental Setup

By adopting the procedure described in [23] the baseline recommender approaches are evaluated by conducting 5-fold cross-validation. The dataset is split randomly into five folds of equal size: four folds as training data and the remaining fold as test data. The process is repeated 5 times such that every fold serves as test data once.

Similar to the procedure described in [23], 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 an 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 is assessed by computing the predicted rating $\hat{r}$ for each movie in the current test set. As described in the following section, the evaluation measure is computed by using the predicted rating $\hat{r}$ and the actual ratings r in the test set. The computation of the evaluation measure is computed for each fold separately and min-max scaling is performed before computing the measures.

D. Evaluation Measures

We utilize two error measures, root mean square error (RMSE) and mean absolute error (MAE), to assess the rating prediction task of our proposed model as well as the benchmark baseline approaches [see Equation 11 and 12]. Hence, min-max scaling is applied to the predicted rating $\hat{r}$ between 0 and 1 to compare the evaluated approaches directly.

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

$$MAE = \frac{\sum_{i=1}^{n}|r_i - \hat{r}_i|}{|n|} \tag{12}$$

where $\hat{r}$ is the predicted rating and r the actual rating as contained in the test set.

E. Results and Discussion

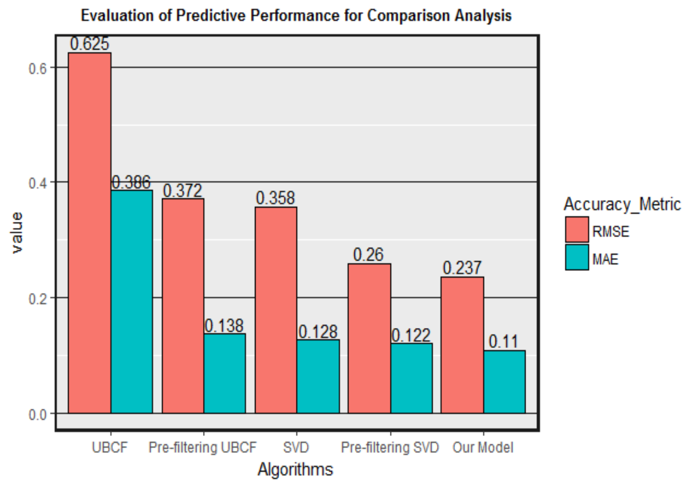
By adopting the evaluation procedure described by Martin et al. [23], the baseline models selected for such comparative analysis with our proposed model are user-based CF recommender system, context-aware CF with pre-filtering, SVD-based recommender system and finally a context-aware SVD-based recommender system with pre-filtering. Table 9 and Fig. 4 stated the results of the rating prediction performance applied to all items. With respect to the rating prediction task, our proposed model clearly outperforms all other approaches by scoring a lower RMSE value of 0.237 and MAE value of 0.110. Following to our model, the SVD-based contextual pre-filtering recommender system achieves higher prediction performance than the rest of the baseline approaches by scoring RMSE value of 0.260 and MAE of 0.052. Lesser to the pre-filtering based SVD approach, the pre-filtering UBCF variant achieves better prediction performance than the remaining recommender systems by scoring RMSE of 0.372 and MAE of 0.132 respectively. The final performance comparison is between the model-based SVD recommender approach and the memory based UBCF recommender approach and the result describes that the SVD recommender system outperforms the UBCF one by scoring RMSE value of 0.358 and MAE value of 0.128. The result we achieved in terms of both RMSE and MAE values proves that rating prediction performance can be improved by integrating context information into the recommendation process. In addition, the slight higher error rate we obtained in both RMSE and MAE result indicated the sparseness of the rating matrix we utilized in this experiment and there are more items (movies in our dataset) a user doesn't show preference to in a given cluster than a user did show preference to.

TABLE IX. EVALUATION OF RATING PREDICTION PERFORMANCE

RECOMMENDER	***RMSE***	***MAE***
UBCF	0.625	0.386
Pre-filtering UBCF	0.372	0.138
SVD	0.358	0.128
Pre-filtering SVD	0.260	0.122
Our Clustered-based Context-Aware Recommendation Model	0.237	0.110

V. CONCLUSIONS

In this research work, we demonstrated a novel approach that extracts relevant contextual features based on Multiple Correspondence Analysis (MCA) feature extraction technique and cluster such low-dimensional contextual features via k-means clustering and incorporates the resulted contextual clusters for the computation of context-aware item recommendation. We identified latent relations between the contextual clusters of a selected item and user's preferences in



such cluster and build a context-aware recommendation model based on these two associations for a rating of items in different possible context clusters. We evaluate the prediction accuracy of our model as well as different benchmark recommendation approaches based on the LDOS-CoMoDa dataset and our proposed model is able to outperform all the baseline approaches significantly. The result we obtained shows that the extraction and clustering of relevant contextual features to explore hidden preferences as well as to the recommendation accuracy is indeed substantial and this is a highly promising research work.

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, pp. 270–274, 2011.

[2] A. Kumaravel and Pallab Dutta, “Application of PCA for Context Selection for Collaborative Filtering,” Middle-East Journal of Scientific Research 20 (1): pp.88-93, 2014.

[3] G. Adomavicius, and A. Tuzhilin, 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 17(6): 734–749, 2005.

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

[5] O. Ante, T. Marko, F. Jurij, K. Andrej, “Relevant Context in a Movie Recommender System: Users’ Opinion vs. Statistical detection,” CARS -2012.

[6] J.P. Benzécri: L’Analyse des données, Tome 1: La taxinomie, Tome 2: l’analyse des correspondances. 2nd edition. Paris: Dunod; 1976.

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

[8] P. S. Bradley, K. P. Bennett, & A. Demiriz, Constrained k-means clustering (Technical ReportMSR-TR-2000-65). Microsoft Research, Redmond, WA, 2000.

[9] C. Ono, Y. Takishima, Y. Motomura, and H. Asoh. Context-Aware Preference Model Based on a Study of. In User Modeling, Adaptation, and Personalization pages 102–113, 2009.

[10] Niu. Donglin, G. Dy, Jennifer and I.J. Michael, Dimensionality Reduction for Spectral Clustering, Proceedings of the 14th International Conference on Artificial Intelligence and Statistics (AISTATS), vol.15, 2011.

[11] G. Adomavicius, R. Sankaranarayanan, S. Sen, and A. Tuzhilin. Incorporating contextual information in recommender systems using a multidimensional approach. ACM Transactions on Information Systems (TOIS), 23(1):103–145, 2005.

[12] M. García-Gil, J. Blanch, M. Comas-Cufí, J. Daunis-i-Estadella, B. Bolíbar, R. Martí, Patterns of stating use and cholesterol goal attainment in a high-risk cardiovascular population: a retrospective study of primary care electronic medical records. J Clin Lipidol. vol.10(1), pp.134–42, 2016.

[13] A. Gediminas and T. Alexander, 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. 17, vol. 6, pp. 734–749, June 2005.

[14] A. Gediminas and T. Alexander, 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, pp. 217–253, 2010.

[15] MJ Greenacre, Correspondence analysis in medical research. Statistical Methods in Medical Research, vol. 1, pp. 97-117, 1992.

[16] L. Baltrunas, B. Ludwig, S. Peer, and F. Ricci. Context relevance assessment and exploitation in mobile recommender systems. Personal and Ubiquitous Computing, pp. 1–20, June 2011.

[17] L. Liu, F. Lecue, N. Mehandjiev, and L. Xu. Using Context Similarity for Service Recommendation. 2010 IEEE Fourth International Conference on Semantic Computing, pages 277–284, Sept. 2010.

[18] L. Lebart, A. Morineau, M. Piron, Statistique exploratoire multidimensionnelle. 3rd edition. Paris, Dunod, 2000.

[19] M. Liao, Y. Li, F. Kianifard, E. Obi, S. Arcona, Cluster analysis and its application to healthcare claims data: a study of end-stage renal disease patients who initiated hemodialysis. BMC Nephrol, vol. 17(4), 2016.

[20] M. Greenacre, Correspondence analysis in practice, London: Academic Press, 1993.

[21] M. Greenacre and J. Blasius, Multiple Correspondence Analysis and Related Methods, Chapman & Hall, 2006.

[22] P. Martin, Z. Eva, and S. Gunther, “Towards a Context-Aware Music Recommendation Approach: What is Hidden in the Playlist Name?” in 15th IEEE International Conference on Data Mining Workshops (ICDM 2015), pp. 1360 – 1365, 2015.

[23] P. Martin, Z. Eva, and S. Gunther, Improving Context-Aware Music Recommender Systems: Beyond the Pre-filtering Approach. In Proceedings of ICMR ’17, June 6–9, 2017, Bucharest, Romania.

[24] F. Alhamid Mohammed, R. Majdi, D. Haiwei, M. Anwar Hossain, E.S. Abdulmotaleb, “Exploring latent preferences for context-aware personalized recommendation systems,” IEEE Transactions on Human-Machine Systems, v.46 n.4, p.615-623, August 2016.

[25] K. Heung-Nam, R. Majdi, A. Abdullah, and E.S. Abdulmotaleb, Folksonomy-based personalized search and ranking in social media services. Information Systems, vol. 37(1), pp. 61-76, 2012.

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

[27] G. Rahul, J.J. Arpit, R. Satakshi, S. Sanjay, “Contextual Information based Recommender System using Singular Value Decomposition”, ICACCI, pp. 2084-2089, 2013.

[28] M. Schedl, A. Vall, K. Farrahi, User Geospatial Context for Music Recommendation in Micro-blogs, In Proceedings of the 37th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR), Gold Coast, Australia (2014).

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

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

[31] D. Solomon and M.Shashi, "A Clustering-Based Context-Aware Recommender Systems through Extraction of Latent Preferences," International Journal of Advanced Research in Computer Science, Volume 9, No. 2, April 2008.

[32] N. Sourial, C. Wolfson, B. Zhu, J. Quail, J. Fletcher, S. Karunananthan, Correspondence analysis is a useful tool to uncover the relationships among categorical variables. J Clin Epidemiol. vol. 63(6), pp.638–46, 2010.

[33] M. Tenenhaus, Statistique. Méthodes pour décrire, expliquer, prévoir, 2nd edition. Paris: Dunod, 2007.

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

[35] K. Yehuda, B. Robert, and V. Chris, Matrix Factorization Techniques for Recommender Systems. Computer 42, vol. 8, pp. 30–37, Aug 2009.

Solomon Demissie Seifu received a bachelor degree in the field of computer science and information technology with distinction in 2006 from Adama Science and Technology University (Ethiopia), then he hold a Masters degree in the field of Information Science in 2010 from Addis Ababa University (Ethiopia), and in 2015 he again received a Master of Technology (M.Tech) degree in Computer Science and Technology from Andhra University, India. He has published 4 papers in international Journals. He worked as Lecturer at the University of Debre Berhan (Ethiopia). Some of his field of teaching includes programming languages like C, C++, Java etc, networking, database and operating systems. His research interests include data mining, machine learning, recommender systems, cloud computing, and software engineering.

Shashi Mogalla is a Professor and Chairperson of Board of Studies of the Department of Computer Science & Systems Engineering, A.U. College of Engineering (A), Andhra University, Visakhapatnam, Andhra Pradesh. She received the AICTE Career Award in 1996, Best Ph.D thesis prize from Andhra University in the year 1994 and AP State Best teacher award in 2016. 13 Ph.D.'s were awarded under her guidance. She co-authored more than 60 technical research papers in International Journals and 50 International Conferences and delivered many invited talks in such academic events. She is a member of IEEE Computational Intelligence group, Fellow of Institute of Engineers (India) and life member of Computer Society of India.. Her current research interests include Data warehousing and Mining, Data Analytics, Artificial Intelligence, Soft Computing and Machine Learning.