# Soft Cosine Gradient and Gaussian Mixture Joint Probability Recommender System for Online Social Networks

Balasubramanian Gayathri Devi[1]*      Venkatasubbu Pattabiraman[1]

[1]*Vellore Institute of Technology, Chennai, Tamil Nadu, India*
* Corresponding author's Email: gayathrimtechcse@gmail.com

**Abstract:** Recommender systems are in recent past becoming more notable in the age of swift evolution of the information technology ensuring e-commerce users' suitable items. In recent years, various thread-based and trust based approaches have been proposed, which improve the precision and recall of recommendation to certain extent. However, these approaches are less accurate than expected when users' preferences evolve over time and when several aspects of the users are involved. To solve these problems, we propose a novel recommendation method called, Soft Cosine Gradient and Gaussian Joint Probability (SCG-GJP) for Online Social Networks (OSN). Specifically, the Local Soft Cosine Clique Clustering (LSCCC) is first used to identify good candidate items to be recommended. Second, using adjacency matrix factorization, probable past preference of individual user and overall preference of whole community is formed to reduce the problem of mean absolute error. Again, the obtained matrix is trained through the stochastic function gradient to rank the most suitable candidates. Finally, Gaussian Mixture Joint Probability Post-filtering model is proposed to provide useful recommendations. The experiments are executed with real-world dataset (DS) and compared with the conventional recommendation methods. Experimental results demonstrate that the SCG-GJP obtains higher accuracy, and lesser error as well as complexity.

**Keywords:** Recommender systems, Soft cosine, Gradient, Gaussian joint probability, Online social networks.

## 1. Introduction

The swift evolution of OSNs has created more awareness for people to interchange events, activities and interests with their friends, colleagues and relatives. Several Service Providers (SPs) advertise via online social sites with the increasing number of users in OSNs. To facilitate users' decisions making, Trust value is applied to select preferred services via recommendation [1].

By applying heterogeneous healthcare information network mining, Thread Recommendation was carried out in [2]. From the Online Health Communities (OHC) data, a heterogeneous healthcare information network was created. As compared to conventional recommendation system that used bipartite graphs which take user nodes and item nodes, heterogeneous healthcare information network maintain the context information of users and threads. To obtain elementary network metrics, thread–thread relationship, and user–user relationship, it extracted features from network [3].

Binary classification was made with resultant features with which thread recommendation was carried out. The similarity rate was enhanced by testing the approaches for different settings and efficient for evaluating user interests. Despite, improvement being observed in terms of similarity rate however the impact of temporal factors was not addressed. To address this issue, in this work a clustering model using Local Soft Cosine Clique is applied to the input DS that also analyzes the users' preferences evolving over time by integrating temporal attributes [4].

High trust value paves the major role as far as service recommendation is concerned in OSNs while users examine certain service. The users in OSNs are inclined to elect services recommended with greater trust value. In contrast, users experience great loss of

property or service when the users get the certain bad services furnished by high trust value. In [5], a Recommendation from High Trust (RHT) was introduced to measure the trust degree of service suggested by high trust value. To achieve this objective, RHT selected the trusted ones according to the ratings based on a high trust value entity. This was performed based on the similarity measure between target user and them. With this, accuracy was found to be higher.

Despite improvement found in accuracy, however, concerns relating to several aspects were not addressed, therefore increasing the complexity while obtaining the recommendations. To provide solution, in this work, Gaussian Mixture Joint Probability Post-filtering model is applied that addressed several aspects related to users that in turn assist users while searching the desired information. Therefore, the difficulty in recommending content was interest to users became a key threat for OSN.

The objective of our paper is to offer user with recommendation for OSN. This is based on both the past preference of the individual user and the preference made by the overall community regarding the item as searched by the user in OSN. We construct a conventional work [2] and [5] to include recommendation on the basis of the preferences. A new model of dynamic recommendation system was introduced which offer user with important items of preference. The proposed method attains user's interests towards item and group them using LSCCC algorithm. Moreover, on the preference matrix, via the Local Clustering Coefficient the accuracy is improved. Our assumption is keeping average similar users in SN and items in each cluster in comparison with global average values will reflect on online experience. Finally, we developed a model for recommending the items by designing a Gaussian Mixture Joint Probability algorithm.

The contributions are as below
1. The intention of different level of user interest towards item for recommendation made to Flipkart users. We construct Clique Undirected graphs for every user constitutes the items. This is depends on utilizing cliques of propositions in users' interest towards item to determine importance level to the user.
2. A Local Soft Cosine Clique model includes user preference and item preference vector in users' interests in preferential items with other social features using adjacency matrix to recommend most suitable candidate items for user.

3. A new Latent Stochastic Gradient Ranking model in OSN is suitable to model the probable past preference of individual user and overall preference of whole community, rather than relying on the users' preference.
4. A Gaussian Mixture Joint Probability Post-filtering model finally forms the main contributing factor to define the set of candidate recommendation.
5. The experiments are conducted to evaluate the performance of SCG-GJP for OSN with real-life DS, Flipkart, Kaggle. The effectiveness of SCG-GJP method is demonstrated in the experimental results.

This paper is arranged as follows. The related work is surveyed in section 2. The details of the SCG-GJP for OSN are explained in section 3. In section 4, the experimental settings are presented. Next, the results and discussion are provided in section 5. Section 6 provides the conclusion.

## 2. Related works

OSN are encountering a swift uptake in both user frequency and frequency of information shared. Therefore, OSN faces several challenges in providing information to user. In [6], a dynamic personalized recommendation system enhances the precision and recall of tweet identification. A comprehensive review and analysis to explore several recommendation approaches was presented in [7]. A hybrid approach of Social and Content aware One-class Recommendation of Papers (SCORP) was introduced in [8].

With the ever-growing increase in the medial consumption in OSN, effective recommendation techniques are found to be demanding to enhance the quality of experience. In [9], a Cluster-based Latent Bias model was designed which monitored users' behavior, location and social contexts, resulting in prediction accuracy. Yet another model with the objective of generating personalized recommendations was designed in [10].

One of the major problems faced by online users is information overloading, hence, several recommender systems have been extensively analyzed to assist user identify relevant information. In [11], activity frequency of user was drawn from power-law distribution with the objective of improving the recommendation accuracy. Two different aspects, like, social similarity and personalized preference were taken into consideration in [12] in obtaining accurate recommendations. Yet another personalized

recommendation model was designed in [13] based on learning object repositories.

A survey of recommender systems for online and mobile social networks (SN) was presented in [14]. One of the successful applications of personalized recommendation model is the collaborative filtering recommendation technology. Owing to, sparse and cold start issues of collaborative recommendation technology, the e-commerce recommendation have certain amount of disadvantages despite being proven to be accurate. In [15], an improved collaborative filtering algorithm based on k-faction was designed with the objective of minimizing calculation time and recommendation speed and accuracy.

But, the current research works has investigated the influence of social interaction infrequently on recommendation performance. This is because of the reason that the recommender systems face higher range of challenges and experiences poor efficiency. To address this issue, a Complete Path Mining (CPM) algorithm was designed in [16] to predict final ratings of items and therefore improvement was observed in terms of recommendation performance. However, user personalities were discarded. A method of group recommendation based on external social trust networks was designed in [17] with the consideration of personality factors. A preliminary evaluation of three use cases for social learning environments was proposed in [18].

With higher amount of data sparsity, complexity in identifying a good recommendation is said to be tedious. To address this issue, a Hybrid Content-based Fuzzy Conformal Recommender (HC-FCR) was investigated in [19] in the domain of movies. With the application of this hybrid method, reliability and accuracy of the system was found to be improved significantly. To enhance recommender system, user item rating matrix was utilized in [20] to obtain social connections between rating patterns to enhance recommendation.

A set of matrix factorization (MF) and nearest-neighbor (NN)-based recommender systems (RSs) was investigated in [21] for discover user SN and group affiliation information for recommendation model depends on social voting. A taxonomy for user modeling on Points Of Interest (POI) was presented in [22].

Although, an enormous research on recommender systems, there is still needs systematic research on analyzing the user preferences evolving over time and considering several aspects. This paper systematically studies the effect of recommendation approach from point of OSN. We can simulate different types of complexity by adding virtual users to the system. We focus on how the individual and overall preference affects the recommender systems performance, including recommendation accuracy and complexity. We introduce three different algorithms which evade unreliable information (via reducing the error) and maintain high recommendation accuracy.

## 3. Methodology

We illustrate a classic scenario where a social recommender system would be sensible. Here, a situation in newbie user of Flipkart, desires to purchase an item, desires to know most recent available brand for items shared by users with social community. To achieve requested information, the user could use search functionality and browse the set of items to retrieve the interesting ones with respect to her/his preferences. The results quality and user satisfaction based on selected search keywords and therefore items which are real interest for user excluded from result set. This would in turn result not only in erroneous information but also the user spending more time and effort. To address this issue, in this work, SCG-GJP for OSN is presented.

### 3.1 System model

Let us assume a recommender system with a set of users '$U=\{u\_1,u\_2,\ldots,u\_m\}$' and set of items '$O=\{o\_1,o\_2,\ldots,o\_n\}$'. For each pair '$\{u\_i,o\_j\}$', a recommender calculate rank '$r\_{ij}$' which attains expected interest of user '$u\_i$' in item '$o\_j$' via knowledge base system and ranking algorithm according to user past preferences and preferences of overall community. Let us assume the set of users and set of items are symbolized via an undirected graph '$G=(V,E)$', with '$V$' denotes the set of vertices (i.e. users in SN) and '$E$' denotes set of edges (i.e. links between users in SN). Fig. 1 given below illustrates the described process in the design of SCG-GJP method for OSN.
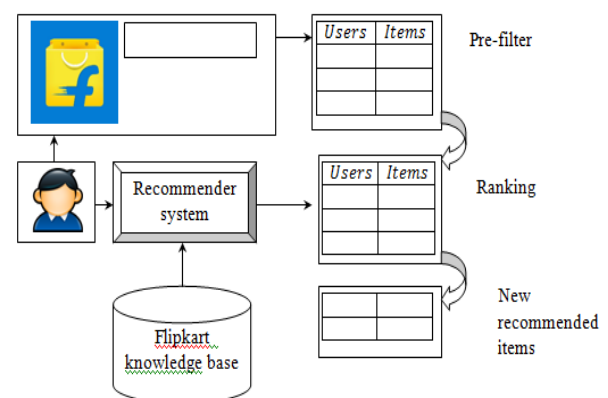


Figure. 1 Process of soft cosine gradient and Gaussian joint probability (SCG-GJP) for online social networks

As illustrated in the figure, to start with a pre-filtering model is performed by applying the Soft Cosine Clique resulting. With this clustering model, both optimality in accuracy and time is said to be achieved by applying a soft cosine similarity between pairs of features (i.e. preferences). With the identified good candidates, ranking model using Latent Stochastic Gradient Recommendation model based on both the past preference of individual user and overall preference of entire community is presented.

Finally, with the most suitable candidates obtained, a post-filtering model using Gaussian Mixture Joint Probability creates the real candidates for recommendation. The overall description of SCG-GJP method for OSN is elaborated in the following sections.

## 3.2 Local soft cosine clique clustering model

In this section, a pre-filtering model based on LSCCC is designed. Let us assume pre-filtering module which selects for every user '$u_i$' a subset '$O_i^c \subset O$' includes of items recommended are good candidates, wherein items normally match user preferences and requires. The pre-filtering is performed in this work by applying LSCCC algorithm. The LSCCC algorithm is given below followed by which, the elaborate description of pre-filtering using LSCCC is provided.

Algorithm 1: Local Soft Cosine Clique Clustering Algorithm

```
Input    : Users 'U=u₁,u₂,…uₙ', Items
'O=o₁,o₂,..oₙ'
Output  : Identify useful candidates to be
recommended 'GC'
1: Initialize number of users and items
2:    Begin
3:       Obtain user preference vector
4:       Obtain item preference vector
5:       Group similar preference pattern using LCC
6:       Obtain soft cosine similarity
7:    Return(good candidates 'GC')
8: End
```

As given in the above LSCCC algorithm, to start with, the number of users and items considered for designing a recommended system is provided. This is performed in our work via an undirected graph model as illustrated in Fig. 2.

Fig. 2 given above illustrates the undirected graph '$G$' comprising of six nodes (i.e. users in SN) and their relationships. Obviously, the nodes '$B$', '$C$', '$D$' and '$E$' represents a '$4 - Clique$' as these nodes are
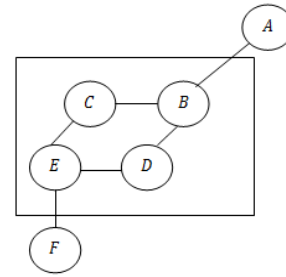


Figure. 2 A Sample undirected graph

associated with each other. Various researchers have presented potential novel methods to obtain enhanced accuracy. The Local Soft Cosine Clique model in SN analysis is designed to improving the accuracy of Flipkart recommendation system.

Let us consider the users '$U$' and items '$O$' are clustered into homogeneous clusters. Choosing the number of user and item clusters is significant. This is because, by choosing minimal clusters, does not achieve preference rate and high number of clusters will reduce the denseness of SN similarity graph, thus resulting in minimal prediction accuracy. The distance factor is measured when performing clustering with optimal accuracy and time, attaining average and measure of dispersion in the 2-dimensional preference vector form, where '$U$' forms preference vector for user '$U$' and '$O$' represents preference vector for item '$O$' are required. Then, the user preference vector '$U_i$' and item preference vector '$O_i$' is expressed as below.

$$U_i = AVG(P_u), \sqrt{\frac{\sum_{i=1}^{n} I_{u,i}\left(P_{u,i} - AVG(P_u)\right)}{\sum_{i=1}^{n}(I_{u,i}-1)}} \qquad (1)$$

$$O_i = AVG(P_o), \sqrt{\frac{\sum_{i=1}^{n} I_{u,i}\left(P_{u,i} - AVG(P_o)\right)}{\sum_{i=1}^{n}(I_{u,i}-1)}} \qquad (2)$$

From the above Eq. (1) and (2), '$AVG(P_u)$' and '$AVG(P_o)$' refers to the '$uth$' users and '$oth$' items average preferences. The preference matrix is clustered in two-planes. User clusters '$n_u$' are attained in the first plane where users with similar preference pattern are grouped by Local Clustering Coefficient (LCC). The LCC quantifies how close its neighbors (i.e. users in SN) are to being a Clique. The LCC is then mathematically formulated as given below.

$$LCC_i = \frac{2[\{e_{ij}: v_i, v_j \in N_i, e_{ij} \in E\}]}{k_i(k_i-1)} \qquad (3)$$

From the above Eq. (3), '$k_i$' represents the number of vertices with '$N_i$' representing the neighborhood for '$k_i$' to '$(k_i - 1)$' links existing between the users. The optimal number of user clusters '$Optimal_U$' and items clusters '$Optimal_O$' are chosen to attain both optimal accuracy and time through average number of similar users in SN and items in each cluster in comparison with global average values. To verify the formed final clusters are not high in the second plane, item clustering is carry out for user clusters and average interactions per user in comparison with global average interactions per user. Using the soft cosine similarity measure, the similar users are measured based on temporal factor in comparison with cosine similarity applied in [2] which does not take temporal factors (i.e. users' preferences evolving over time). Then, the soft cosine similarity '$Soft_{Cosine}$' measure for two different vectors '$p$' and '$q$' is measured as given below.

$$GC = Soft_{Cosine}(p,q) = \frac{\sum_{i,j=1}^{n} Sim_{ij} p_i q_j}{\sqrt{\sum_{i,j=1}^{n} Sim_{ij} p_i p_j}\sqrt{\sum_{i,j=1}^{n} Sim_{ij} q_i q_j}}$$

(4)

From the above Eq. (4), the soft cosine similarity measure is evolved for two different vectors by considering the similarity of features '$Sim_{ij} = similarity(feature\ i, feature\ j)$'. Here, features represent the users' preferences. Finally, based on preferences, the adjacency matrix of user similarity is generated is shown in Table 1 for six different users.

Table 1 depicts the adjacency matrix which illustrates the relationship between users. When the value of two users is '1', which denotes they have similar preferences, or else, they are not similar in preferences to other. Based on the Local Soft Cosine Clique model, the adjacency matrix cluster users into different groups. Upon completion of this process, several clusters of users are formed with higher accuracy rate and optimal time.

### 3.3 Latent stochastic gradient ranking model

With the obtained good candidates for recommendation, the second step is to design a recommendation model. In this work, Latent Stochastic Gradient Ranking model is designed with the objective of reducing the training time in ranking.

Let us assume a ranking module which assigns user '$u_i$' rank '$r_{ij}$' for every candidate item '$o_j$' with the recommendation technique x, which is employed in diverse ways items' features and users' preferences, user past behavior and user feedbacks. The ranking

Table 1. Adjacency matrix user preferences

|  | User$_1$ | User$_2$ | User$_3$ | User$_4$ | User$_5$ | User$_6$ |
|---|---|---|---|---|---|---|
| User$_1$ | 1 | 0 | 0 | 0 | 1 | 1 |
| User$_2$ | 0 | 0 | 1 | 1 | 1 | 0 |
| User$_3$ | 0 | 1 | 0 | 0 | 1 | 1 |
| User$_4$ | 1 | 0 | 0 | 1 | 0 | 0 |
| User$_5$ | 0 | 1 | 0 | 0 | 1 | 0 |
| User$_6$ | 0 | 1 | 1 | 0 | 1 | 1 |

of set of items '$O$' embedding in collaborative learning context their features is the main intention of this stage.

Here, probable past preference of individual user and overall preference of whole community is considered to provide useful recommendations using adjacency matrix factorization. The Latent Stochastic Gradient algorithm is given below followed by which, the elaborate description of recommendation model based on ranking is provided.

Algorithm 2: Latest Stochastic Gradient Algorithm

---
**Input:** good condition "GC"
**Output:** Most suitable condition 'Ai, Bj', 'WC(A), WC(B)'
1:**Begin**
2:    **Initialize** good candidates 'GC'
3:    Measure sum of errors between the actual rating and predicted rating to obtain the loss
4:    Measure stochastic function gradient for customized ranking for each individual user
5:    Evaluate user feature matrix and item feature matrix based on past preference
6:    Evaluate user feature matrix and item feature matrix based on overall performance of whole community.
7:    **Return**(Most Suitable Candidates 'Ai, Bj', 'WC(A), WC(B)')
8:**End**

---

As given in the above Latent Stochastic Gradient algorithm, the main objective remains in minimizing the mean absolute error (i.e. recommendation error) involved during the design of a recommendation model. This is achieved in our work based on the design of adjacency matrix factorization. The essence of adjacency matrix factorization remains in splitting the rating matrix into two matrices, the first one denoting the user characteristics and the second one denoting the item features based on the proposition. Finally, the inner product of each row and each column results in the corresponding rating.

$$R_{i*j} = A_{i*k} * B_{k*j}$$

(5)

From Eq. (5), '$A_{i*k}$' denotes the relationship between '$i$' users and '$k$' propositions. '$B_{k*j}$' denotes relationship between '$k$' propositions and '$j$' items, with '$k$' representing the latent factors. Hence, the main objective of recommendation algorithm based on adjacency matrix factorization is to solve '$A_{i*k}$' and '$B_{k*j}$'. Besides, a loss function, '$Loss$' indicates the sum of errors among actual ratings and predicted ratings is expressed as given below.

$$Loss = \sum_{i=1}^{n}\sum_{j=1}^{m} R_{ij} - \sum_{k=1}^{f} A_{ik}B_{kj} \quad (6)$$

From the above Eq. (6), the loss function '$Loss$' is obtained based on the '$A_{ik}$' representing the '$kth$' feature of '$ith$' user and '$B_{kj}$' representing the '$kth$' feature of '$jth$' item with respect to the corresponding rating '$R_{ij}$' respectively. Then, to obtain the probable past preference of individual user and overall preference of whole community, a Stochastic Gradient Descent (SGD) function is applied with the above occurred loss function. The purpose of applying the SGD function is that only few predictions obtained from several users (i.e. whole community) are selected randomly instead of the whole data set for single iteration. The model uses only a single sample, a batch size of one to perform single iteration, with the sample being shuffled in a random manner for performing the iteration. The stochastic function gradient for customized ranking for each individual user is expressed as given below.

$$\frac{\partial Loss}{\partial A_{ik}} = \left[R_{ij} - \sum_{k=1}^{f} A_{ik}B_{kj}\right]B_{kj} + \alpha A_{ik} \quad (7)$$

$$= e_{ij}B_{kj} + \alpha A_{ik} \quad (8)$$

$$\frac{\partial Loss}{\partial B_{kj}} = \left[R_{ij} - \sum_{k=1}^{f} A_{ik}B_{kj}\right]A_{ik} + \alpha B_{kj} \quad (9)$$

$$= e_{ij}A_{ik} + \alpha B_{kj} \quad (10)$$

From Eqs. (8) and (10), the optimal user feature matrix '$A_{ik}$' and item feature matrix '$B_{kj}$' based on the past preference of individual user is expressed as below.

$$A_{ik} = A_{ik} + \lambda\left(2e_{ij}B_{kj} - \alpha A_{ik}\right) \quad (11)$$

$$B_{kj} = B_{kj} + \lambda\left(2e_{ij}A_{ik} - \alpha B_{kj}\right) \quad (12)$$

In a similar manner, the overall preference of whole community is mathematically expressed as given below.

$$WC(A_{ik}) = 1 - A_{ik} \quad (13)$$

$$WC(B_{kj}) = 1 - B_{kj} \quad (14)$$

With the above past preference of individual user obtained from (11), (12) and overall preference of whole community from (13), (14), recommendations are provided based on the user interactions and generated items via propositions from flipkart.

### 3.4 Gaussian mixture joint probability post-filtering model

Using the Context Gaussian Mixture function, a post-filtering model is attaining the set of real candidates for recommendation with the obtained most suitable candidates using SGD function. The context is denoted by Gaussian Mixture function using dimensions past preference of individual user and overall preference of whole community. The Gaussian Mixture Joint Probability algorithm is given below followed by which, the elaborate description of the post-filtering is provided.

Algorithm 3: Gaussian Mixture Join Probability Algorithm

**Input**: suitable candidates 'Ai, Bj', 'WC(A), WC(B)'
**Output**: Set of items (candidates) recommended
1:**Initialize** suitable candidates 'Ai, Bj', 'WC(A), WC(B)'
2: **Begin**
3: Obtain conditional probability of the rating based on past preference and overall preference
4: Obtain set of candidate recommendations
5:**End**

As given in the above Gaussian Mixture Joint Probability algorithm, for each suitable candidate obtained based on the Latent Stochastic Gradient algorithm, the objective of the algorithm remains in generating a set of candidate items with minimum complexity. With this objective, using the Gaussian mixture model with the conditional probability of the rating based on past preference and overall preference. This is mathematically expressed as given below.

$$Prob\left(r|A_i, B_j, WC(A_i), WC(B_j)\right) = O_i^C = \\ Prob\left(z_k|A_i, B_j\right)Prob\left(z_k|WC(A_i), WC(B_j)\right) \quad (15)$$

From the above Eq. (15), '$Prob\left(r|A_i, B_j, WC(A_i), WC(B_j)\right)$' corresponds to the joint probability of both the past preference,

overall preference '$Prob(z_k | A_i, B_j)$' denotes the probability that the user '$A$' with preferential item '$B$' belong to the group '$z_k$' and '$Prob(z_k|WC(A_i), WC(B_j))$' represents that the rating of each user forming overall preferences in group '$z_k$' obeys Gaussian distribution that denotes the conditional probability of the ratings on user '$A$' with preferential item '$B$' from the group '$z$'. Consider, if user '$u_i$' is currently interested in target item '$o_j$', then the set of candidate recommendations are formalized as below.

$$O_{ij}^F = \{o_i \in O_i^C\} \cup \{o_i \in NNQ(o_j, O_i^C)\} \quad (16)$$

In this manner, a new set '$O_{ij}^F \subseteq O_i^c$' is produced based on the past preference of individual user and overall preference of whole community with minimum complexity. Ultimately, depending on the applications, recommended items are said to be organized in consonance with the supplementary constraints.

## 4. Experimental settings

Recommender Systems is considered to be very complicated. Hence, their evaluation is also considered to be the most challenging task. The research efforts on evaluating recommender system were focused on precision and recall [2]. In recent times, researchers investigating the issues of evaluating users' preferences with trust to evaluate recommender systems [5]. However, only with a simple aspect, recommendation system cannot be designed. Starting from these examinations and on the basis of recent conventional works, an evaluation of recommended algorithm is performed in terms of complexity, accuracy and mean square error with number of users and compared with existing [2] and [5].

### 4.1 Dataset

We employed a subset of Yahoo Flipkart products i.e., 20,000 extracted from https://www.kaggle.com/PromptCloudHQ/flipkart-products. This is pre-crawled DS considered as subset of bigger **DS (more than 5.8 million products)** which was constructed via extracting data from Flipkart.com, a foremost Indian e-Commerce store. It consists of 15 columns.

### 4.2 Evaluation metrics

In this paper, computational complexity is used

Table 2. Description of parameters

| S1.No | Attributes | Data type |
|---|---|---|
| 1 | Uniq id | String |
| 2 | Crawl timestamp | Data |
| 3 | Product url | String |
| 4 | Product name | String |
| 5 | Product category tree | String |
| 6 | Pid | String |
| 7 | Retail price | Numeric |
| 8 | Discount price | Numeric |
| 9 | Image | String |
| 10 | Is FK Advantage product | Boolean |
| 11 | Description | String |
| 12 | Product rating | Numeric |
| 13 | Overall rating | Numeric |
| 14 | Brand | String |
| 15 | Product specifications | String |

for evaluating the proposed method. The complexity involved in designing any application is referred to as the computational complexity. In this work, a recommendation system for OSN is designed and hence, the computational complexity is defined as the complexity involved in designing a recommendation system. It is measured according to the clustering applied with respect to the number of users requested for purchasing a specific item (i.e. products) and the time involved in identifying the optimal clusters.

$$CC = \sum_{i=1}^n U_i * Time(GC) \quad (17)$$

From the above Eq. (17), the computational complexity '$CC$' is measured according to the time consumed in identifying the optimal number of item clusters with respect to the number of users '$U_i$' considered for experimentation. Here, the optimal number of item clusters are generated based on the soft cosine similarity function '$GC$' and is measured in terms of milliseconds (ms).

In addition, mean absolute error and recommendation accuracy are also extensively used metrics for measuring the recommendation method. The evaluation metrics are averaged over all users. Next, Mean Absolute Error (MAE) or the recommendation error is used for measuring the performance of the proposed method. The MAE measure for the user '$u$' is calculated as follows

$$MAE = \frac{\sum_{i=1}^n [RP_{uo} - PP_{uo}]}{n} * 100 \quad (18)$$

From the above Eq. (18), the mean absolute error '$MAE$' is measured according to the real preferences '$RP$' and predicated preferences '$PP$' of the item '$o$' for user '$u$', with '$n$' representing the total number of preferences made by the recommendation method.

The recommendation method is evaluated by measuring the recommendation accuracy. Recommendation accuracy is the ratio of number of samples correctly recommended by the system for user query to the selection of items to the total sample size for a given test data set. The recommendation accuracy is measured as given below.

$$RA = \frac{S_{correct}}{S_{test}} * 100 \qquad (19)$$

From the above Eq. (19), the recommendation accuracy '$RA$', is made based on the samples correct '$S_{correct}$' and the total test samples '$S_{test}$' respectively. It is measured in terms of percentage (%).

### 4.3 Results of the experiments

#### 4.3.1. Parameter settings

Table 2 given above shows the details about the parameters used in all methods, SCG-GJP for OSN, Thread Recommendation [2] and RHT [5] respectively, including their meanings and data types.

#### 4.3.2. The influence of the parameters

First, the impact of the parameter changes in the computational complexity of each step on the proposed method is analyzed and find out their optimal configuration by setting up multiple sets of different number of users. Computational complexity refers to the complexity involved in analyzing the number of users requested for purchasing a specific item. Fig. 3 shows the performance of the computational complexity measured in terms of milliseconds (ms) by varying the number of users in the range of 25 to 250.

As illustrated in the above figure, 250 different numbers of users are used for conducting experimentation using the SCG-GJP, Thread Recommendation and RHT respectively. Here, the computational complexity is measured for 250 different numbers of users. With '25' users requested for recommendation of an item while performing a purchase, the computational complexity for single user to obtain the recommendation being '$0.53ms$', the ove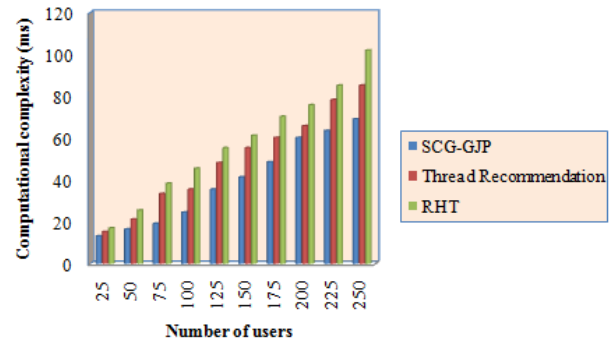rall computational complexity was observed to be '13.25$ms$' using SCG-GJP. In a similar manner, with '25' users requested for recommendation of an item while performing a purchase, the computational complexity for single user to obtain the recommendation being '$0.61ms$', the overall computational complexity was observed to be

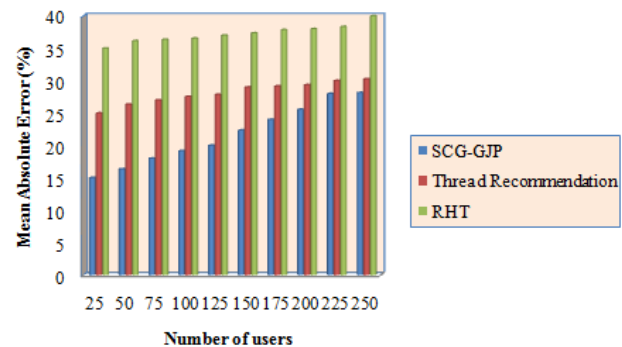

Figure. 3 Impact of computational complexity



Figure. 4 Impact of mean absolute error

'15.25$ms$' using Thread Recommendation and '17$ms$' using RHT.

From this analysis it is inferred that the complexity involved measuring the optimal number of item clusters was found to be minimum using the SCG-GJP. This is because of the application of Gaussian Mixture Joint Probability algorithm. By applying the Gaussian Mixture Joint Probability algorithm, a set of candidate items were generated using the Gaussian mixture model. With this model, the conditional probability of the rating was used to analyze the complexity according to both the past preference and overall preference. In this way, the candidate items were generated in a minimum amount of time using SCG-GJP method by 23% when compared to [2] and 33% when compared to [5] respectively.

The influence of recommendation error is evaluated with numbers of user's ranges from 25 to 250 with Flipkart Products KaggleDS. Fig. 4 shows the mean absolute error (i.e. recommendation error) for different numbers of users.

Fig. 4 given above shows the mean absolute error. The mean absolute error refers to the wrong prediction made by the system or the method in use. From the figure it is inferred that the number of users is directly proportional to the mean absolute error. With higher number of users requested for purchase of an item, the system gets overloaded with the set of
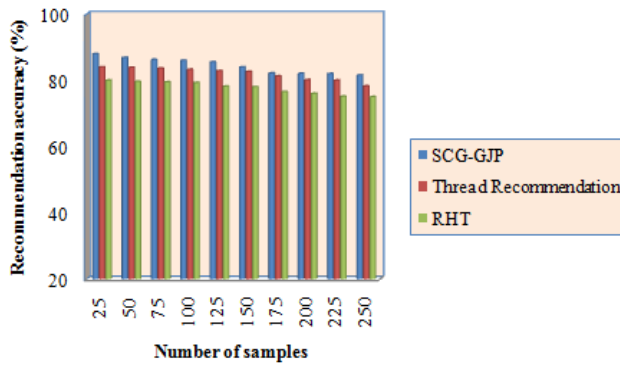
Figure. 5 Impact of recommendation accuracy

items to be recommendation. Hence, the number of users. However, as shown in Fig. 4, the mean absolute error is found to be reduced using SCG-GJP method when compared to Thread Recommendation and RHT. This is evident from the sample calculations. With '20' total number of preferences made by the recommendation method, real preferences being '18' and predicated preferences being '15', the mean absolute error using SCG-GJP method was found to be '15%'. In a similar manner, with '20' total number of preferences made by the recommendation method, real preferences being '18' and predicated preferences being '13', the mean absolute error using Thread Recommendation was found to be '25%' and '35%' using RHT respectively. From the simulation, it can be seen that the recommendation error is comparatively lesser using SCG-GJP method. This is because of the application of Latent Stochastic Gradient algorithm. By applying this Latent Stochastic Gradient algorithm, with an adjacency matrix factorization, two matrices, one for user characteristics and second for item features are acquired separately. Finally, by utilizing the inner product of each row and column to obtain the corresponding rating, most suitable candidates are generated. With this, the recommendation error of mean square error is reduced using the SG-GJP method by 24% when compared to [2] and 42% when compared to [5].

The performance of recommendation method is evaluated by measuring the recommendation accuracy. Experiments are conducted to prove that the recommendation method is more efficient than the conventional works with different numbers of sample. The comparison results are shown in Fig. 5.

The results in Fig. 5 given above clearly indicate that the recommendation accuracy is higher using the SCG-GJP than the other conventional works. Here, x axis denotes the number of samples ranges from 25 to 250. Here, samples refer to the number of product preferences queried by different users at different time intervals. To conduct fair comparison, similar

numbers of samples are used for SCG-GJP and the state-of-the-art methods. With '25' samples considered for experimentation and '22' samples correctly recommended, the overall recommendation accuracy using SCG-GJP method was found to be '88%'. In a similar manner, with '25' samples considered for experimentation and '21' samples correctly recommended, the overall recommendation accuracy was found to be '84%' using Thread Recommendation and '80%' using RHT. From this it is inferred that the recommendation accuracy is found to be better using SCG-GJP method when compared to [2] and [5]. This is because of the application of LSCCC algorithm. By applying this algorithm, first, useful candidates to be recommended was identified via soft cosine similarity. Next, with this useful candidates recommended, ranking of the candidate items were performed via Latent Stochastic Gradient. This in turn recommended the most suitable candidates. Finally, with this the set of items were recommended for OSN by applying Context Gaussian Mixture function. With this the recommendation accuracy using SCG-GJP method was improved by 3% when compared to [2] and 9% when compared to [5].

## 5. Conclusion

A novel recommendation method for OSN is introduced in this paper. Based on soft cosine similarity, a constrained similarity measure is proposed to lessen the complexity in obtaining the recommendation. Besides, a novel multiple steps method is presented to increase recommendation accuracy with probable past preference of individual user and overall preference of whole community while performing recommendation model. The proposed method first constructs the user-based vector and then the item-based vector and then uses the LCC to perform ranking to minimize the dimension of rating matrix. Finally, Gaussian mixture model using the conditional probability of rating based on past preference and overall preference is proposed to obtain the final set of recommendations, algorithm, aiming at improve the accuracy of recommendation. The experiments are conducted with Flipkart products Kaggle dataset to show the SCG-GJP method attains greater recommendation accuracy with minimum complexity and error compared to the conventional methods.

# References

[1] E. Chirchi and K. Digambar, "Modified Circular Fuzzy Segmentar and Local Circular Encoder to iris Segmentation and Recognition", *International Journal of Intelligent Engineering and Systems*, Vol. 10, No. 2, pp. 183-192, 2017.

[2] C. C. Yang and L. Jiang, "Enriching User Experience in Online Health Communities through Thread Recommendations and Heterogeneous Information Network Mining", *IEEE Transactions on Computational Social Systems*, Vol. 5, No. 4, pp.1049 – 1060, 2018.

[3] A.V. Haridas, R. Marimuthu, and V.G. Sivakumar, "A critical review and analysis on techniques of speech recognition: The road ahead", *International Journal of Knowledge-based and Intelligent Engineering Systems*, Vol. 22, No. 1, pp. 39-57, 2018.

[4] S. Shekhar, D.K. Sharma, and M.M.S. Beg, "An effective cybernated word embedding system for analysis and language identification in code-mixed social media text", *International Journal of Knowledge-based and Intelligent Engineering Systems*, Vol. 23, No. 3, pp. 167-179, 2019.

[5] Y. Xiao, Q. Pei, X. Liu, and S. Yu, "A Novel Trust Evaluation Mechanism for Collaborative Filtering Recommender Systems", *IEEE Translations and Content Mining*, Vol. 6, No. 1, pp. 70298-70312, 2018.

[6] S. Khater, D. Gracanin, and H.G. Elmongui, "Personalized Recommendation for Online Social Networks Information: Personal Preferences and Location-Based Community Trends", *IEEE Transactions on Computational Social Systems*, Vol. 4, No. 3, pp.104-120, 2017.

[7] A. Anandhan, L. Shuib, M.A. Ismail, G. Mujtaba, "Social Media Recommender Systems: Review and Open Research Issues", *IEEE Access*, Vol.6, No. 1, pp.15608-15628, 2018.

[8] G. Wang, X. He, and C.I. Ishuga, "Social and content aware One-Class recommendation of papers in scientific social networks", *PLOS ONE*, Vol. 12, No. 8, pp. 1-30, 2017.

[9] C. Wu, Y. Zhang, J. Jia, and W. Zhu, "Mobile Contextual Recommender System for Online Social Media", *IEEE Transactions on Mobile Computing*, Vol. 16, No. 12, pp. 3403-3416, 2017.

[10] C. Zhang, J. Liu, Y. Qu, T. Han, X. Ge, and A. Zeng, "Enhancing the robustness of recommender systems against spammers", *PLOS ONE*, Vol. 13, No.11, pp. 1-14, 2018.

[11] D.-B. Chen, G.-N. Wang, A. Zeng, Y. Fu, and Y.-C. Zhang, "Optimizing Online Social Networks for Information Propagation", *PLOS ONE*, Vol. 9, No. 5, pp. 1-8, 2014.

[12] D-C. Nie, Z-K.Zhang, J-L.Zhou, Y.Fu, and K. Zhang, "Information Filtering on Coupled Social Networks", *PLOS ONE*, Vol.9. No.7, pp. 1-15, 2014.

[13] C. K. Pereira, F. Campos, V. Ströele, J.M.N. David, and R. Braga, "BROAD-RSI – educational recommender system using social networks interactions and linked data", *Journal of Internet Services and Applications*, Vol. 7-9, No. 1, pp. 1-28, 2018.

[14] M.G. Campana, and F. Delmastro, "Recommender Systems for Online and Mobile Social Networks: A survey", *Online Social Networks and Media,* Vol. 3-4, No. 1, pp. 75-97, 2017.

[15] X. Li and D. Li, "An Improved Collaborative Filtering Recommendation Algorithm and Recommendation Strategy", *Mobile Information Systems,* Vol. 2019, No. 1, pp.1-11, 2019.

[16] Y. Li, J. Liu, and J. Ren, "Social recommendation model based on user interaction in complex social networks", *PLOS ONE*, Vol.14, No.7, pp.1-17, 2019.

[17] G. Fang, L.Su, D. Jiang, and L. Wu, "Group Recommendation Systems Based on External Social-Trust Networks", *Wireless Communications and Mobile Computing*, Vol. 2018, No. 1, pp.1-11, 2018.

[18] S. Karthick, "Semi Supervised Hierarchy Forest Clustering and KNN Based Metric Learning Technique for Machine Learning System", *Journal of Advanced Research in Dynamical and Control Systems*, Vol. 9, No. 18, pp. 2679-2690, 2017.

[19] S. Ayyaz, U. Qamar, and R. Nawaz, "HCF-CRS: A Hybrid Content based Fuzzy Conformal Recommender System for providing recommendations with confidence", *PLOS ONE*, Vol.13, No.10, pp. 1-30, 2018.

[20] L. Guo, Y-F. Wen and X-H. Wang, "Exploiting Pre-Trained Network Embeddings for Recommendations in Social Networks", *Journal of Computer Science and Technology*, Vol. 33, No. 4, pp 682–696, 2018.

[21] X. Yang, C. Liang, M. Zhao, H. Wang, H. Ding, Y. Liu, Y. Li, and J. Zhang, "Collaborative Filtering - Based Recommendation of Online Social Voting", *IEEE Transactions on Computational Social Systems*, Vol. 4, No. 1, pp.1-13, 2017.

[22] S. Liu, "User Modeling for Point-of-Interest Recommendations in Location-Based Social Networks: The State of the Art", *Mobile*

*Information Systems*, Vol. 2018, No. 1, pp.1-13, 2018.