IJCTA, 9(10), 2016, pp. 4221-4231

Analysis of location based recomended algorithm

***Dr. K.Thangadurai **M. Venkatesan**

Abstract : One of the most influential weapons in this digital world is Recommender systems. They normally afford explanation to their recommendations so that the web users can find their products, people and even their friends who are missing in these social communities. There are a variety of methods and approaches which have been implemented in this recommender system until now. Content-based and collaborative approaches are the most extensively used approaches . We may study these approaches in order to have the most personalized approaches and to have the best recommendations for the end users. We always find some trouble in the relationship between job seekers and job portal companies. The reason is that mostof the recommendations are inappropriate for the job seekers. This also may be due to the limitation of the location and the date and time of the interview. This paper presents the analysis of recommendations algorithm on the basis of its location and attribute recommender. This paper focuses on CLAF, IFC, PCLAF and CADC by making use of these algorithms for job recommendation.

Keywords : Recommender system, CLAF, IFC, PCLAF, Location Based Recommendation, Job Recommendation.

1. INTRODUCTION

Initially Recommender systems were defined as ones where ‘‘people provide recommendations as inputs, which the system then aggregates and directs to appropriate recipients’’ [17]. The obvious main objective of the existing recommender systems is to direct the user to objects which are useful/interesting (refer figure 1). Due to this, evaluation of recommender systems implies the review of how much of this goal has been achieved.

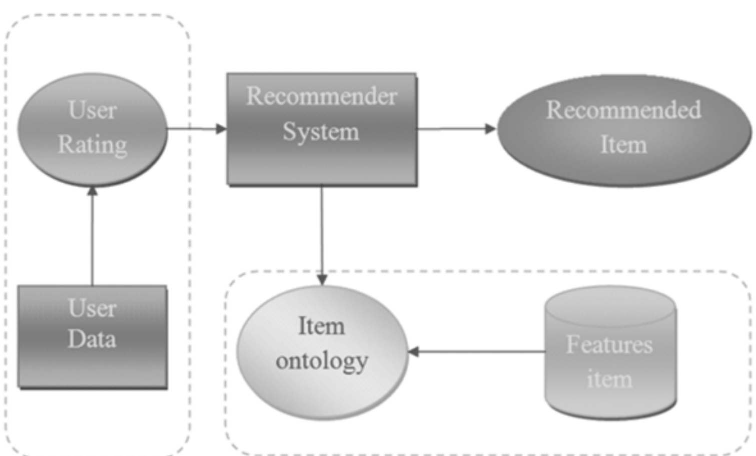

Fig. 1. General architecture of location based recommended system.

* Asst. Professor & Head PG and Research Department of Computer Science, Government Arts College (Autonomous), Karur-639005, Tamilnadu, India. Email: ktramprasad04@yahoo.com,

** PG and Research Department of Computer Science, Government Arts College (Autonomous), Karur-639005, Tamilnadu, India. Email: mpkvenkat@gmail.com

Recommender system normally affords explanation to their recommendations so that the users can choose their products, activities or even friends. It is the duty of recommender systems to turn data on users and their preferences into predictions of the possible future likes and interests of the users. When an explanation is received by a user, he can accept a recommendation more easily. The reason is that the system provides lucidity to its recommendations (which follows most of recommender algorithms). The Human Style, Feature Style and Item Style approaches are followed by the most traditional approaches .This is in fact a simple approach but it can be realized in a number of ways. The reason is that the field of recommendation lacks general ''first principles'' and only from this one could infer the correct way to recommend. For example [7], How can one measure the user similarity and assess its uncertainty to the level best? How can one aggregate different opinions from various users? How can one handle users who have little information available? Should one trust all data equally or can one find out reckless or intentionally misleading opinions? These and alike problems also arise when techniques are more sophisticated than those based on user similar use.

1.1. Definitions and Basic Factors

CF algorithms are among the most popular methods in recommender systems and in this section, we recognize the chief factors that critically affect all CF algorithms. There are three stages in the basic operations of the CF process and our analysis concentrates on the same.

- **Stage 1 :** Formation of the user or item neighbourhood with objects of similar ratings and behaviour.
- **Stage 2 :** Generation of a top-N list with algorithms that construct a list of best item recommendations for a user
- **Stage 3 :** Quality assessment of the top-N list.

1.2. Definitions and Basic Factors

Table 1 : Factors that affect CF algorithms

Factor name	*Short description*	*Stage*
Sparsity	Limited percentage of rated products	1
Scalability	Computation increase by the number of users and items	1
Train/test data size	Data are divided into training and evaluation or test	1, 3
Neighborhood size	Number of neighbors used for the neighborhood formation	1
Similarity measure	Measures that calculate proximity of two objects	1
Recommendation list size	Number of top-N recommended items	2
Recommendation list creation	Algorithms for the top-N list generation	2
Positive rating threshold	Positive and negative ratings segregation	2, 3
Evaluation Metrics	Metrics that evaluate the quality of top-N list	3
Setting a Baseline method	A simple method against which performance is compared	3
Past/future items	The segregation between a priori known and unknown items	3

Besides, there are a few more factors which have been identified. For instance, the impact of subjectivity during the rating or issues related to the preprocessing of data [3,4,13].However, these factors are not examined by us because their effect is less easily determinable

1.3. From OSNs to LBSNs

People are led to new solutions in various fields due to Technological progress. In the case of OSNs, there is a rapid evolution of technologies. This enables it to acquire geo -location data that lead the transition from traditional social networking services to Location-based ones. With the evolution of smart phone and the integration of technologies like 3G or 4G, Wi-Fi and GPS in it, it has become easy to acquire geo-position data. As a result, every individual who owns a smart phone can access the use of Location-based services. On the basis of a user's preference, these LBSN systems can also serve as location recommendations. In a location recommendation, the LBSN system can suggest the place when a user wants to do something. In Job recommendation, the LBSN precedes a step ahead.

Adding location dimension is a task which is more challenging. Traditional graphical user interface of OSNs is considered old for LBSNs. The necessity for the presence of a map in the GUI of the LBSN is considered an inevitability, in order to envisage users' movements in the geo-locational dimension. Besides, an OSN consists of one kind of nodes (unipartite graph), while an LBSN (except user-user graph) consists of a bipartite (user-location) or even tripartite graph (user-location-job).

1.4. Definitions and Services of LBSNs

According to the report of Li and Chen, [6] LBSNs allow the users to perceive where their friends are, to find out location-tagged content within their social graph, and to meet others nearby. Zheng and Zhou [22] asserted that LBSNs comprise the new social structure made up of individuals joined by the interdependency drawn from their locations in the physical world as well as their location-tagged job content, like developer, java developer, system admin and system analysis . As said by Wikipedia, LBSNs are a kind of social networking in which geographic services and capabilities such as geo-coding and geo-tagging are used to facilitate additional social dynamics.

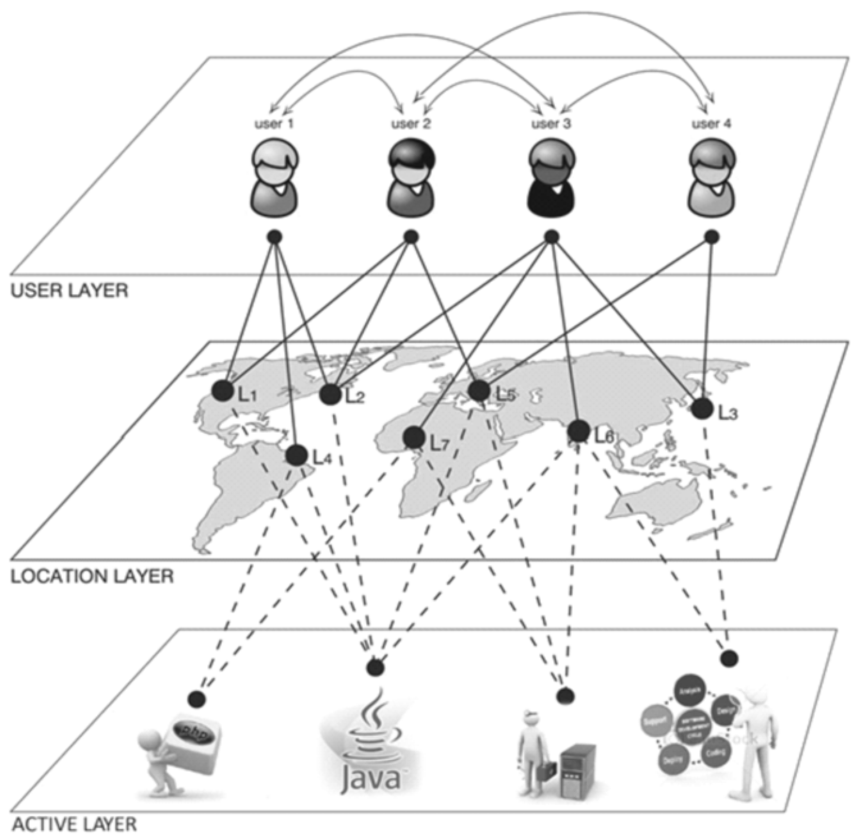

Fig. 2. Three layers of user, location and job content.

Fig. 2 shows that users can visit locations in the real world and can provide geo-tagged information job content (*e.g*. developer, java developer, system admin etc). Fig. 2 specifically portrays three layers, namely, the user, location, and job content layers. What is obvious is that someone can exploit information from each layer independently to influence recommendations. For example, the geographical distance (*i.e*. Euclidean distance) between each pair of places in the location layer can easily be calculated. Moreover, the similarity among users can also be calculated. This can be done on the basis of the social network that exists in the user layer. With regard to the job content layer, similarity among the information objects can be computed (*i.e*. PHP, tags etc.) based on their metadata including the ternary relation among entities (*i.e*. user, location, job), which goes through all layers.

If this abundant contextual information is acquired, LBSNs can improve the quality of services on: (1) generic (non-personalized) recommendations of social events, user, locations, and job, (2) personalized recommendations of social events, user, locations, and job and (3) user and group mobility behaviour modeling and community discovery. The rest of the analysis briefly discusses the research work for each of the categories of LBSNs services which are mentioned above.

1.5. Generic Recommendations

Generic Recommendations calculate the similar recommendation list (user, locations, and job etc.) for all users despite the personalized preferences of every individual user. The recommender systems which are based on counting the frequencies of occurences or co-occurences of any dimension which is given are the simplest ones.

1.6. Personalized Recommendations

The personalized recommender systems depend on the earlier history of users. After acquiring the past history, they associate them with other users with alike preferences and propose them new user, locations and job. To say specifically, a personalized recommender makes use of the time that someone has visited a location. The three approaches that have become known in the context of recommender systems are collaborative filtering (CF), content-based Filtering (CB) and hybrid methods.

The locations, activities and events in a city are recommended to the target user through CF methods. Location-based CF algorithm makes use of the similarities in the locations so that the neighborhood of nearest users can be formed and this reduces the problem of scalability. A pitfall location-based CF is the cold start problem: new users have performed only a few visits.

It is assumed by CB methods that each user functions independently and due to this , it is able to utilize the information which is derived only from location features. For instance, there may be features like cuisine and cost in a restaurant. If a user prefers a Chinese cuisine in her profile, then she will be presented with the Chinese restaurants. It appears that these systems have certain limitations that they do not consider the preferences of other people. To say specifically, there are a set of attributes that describes the location. It makes use of those and recommends other locations which are similar to those that exist in the user's profile. The cold start problems that are faced by CF methods for new locations and new users are reduced through this way. Nevertheless, there is a drawback of CB and it is that there is no diversity in the location and activity recommendations.

There is a way of handling short comings and it is the combination of social with geographical data. It is possible when we take only one type of data into consideration.

2. RECOMMENDER SYSTEMS FOR LBSNS

Recommender systems may utilize the current location of the users as a supporting source. By doing so, their recommendations about places or activities that a target user may be interested in can be improved. Normally, recommenders in LBSNs function under the same structure as in traditional OSNs. There is only one main difference and it is that, in LBSNs, the user's location history is the fundamental algorithmic input used to generate a recommendation. Particularly, the recommendation algorithm is struggling to find out likeness by using either the locations visited by the users or the location of the current users.

2.1. Recommendation Types

Users provide location data and on this basis, LBSN providers can yield three distinct kinds of recommendations and they are User, Location and Job recommendation. Each one of these types is introduced below.

Location or Point of Interest (POI) Recommendation

Let us assume that a user is in an unknown place and she needs information regarding sight seeing places or the place where she can get food. Location or POIs recommendation can offer supportive direction to a user in order to visit the places of interest. Ye et al.[18] argues that an important role in user's check-in behaviour is played by the geographical influence among POIs and it is modeled by means of a power law distribution. Moreover, a unified POI recommendation framework is proposed by him and it fuses user's preference to a POI with social and geographical influence. Saez-Trumper et al. [12] has done one more very recent study in which he has illustrated that when we recommend places that are closest to a user's geographic center of interest (*e.g.* user's home), it produces recommendations and they are very accurate like item-based collaborative filtering algorithm.

3. LOCATION BASED JOB RECOMMENDATION

Recommender systems are influential devices and they offer explanations to the job seekers. It is an important work of recommender system to find out the chances as per the users' likes and interest. This main analysis of this paper on the recommender system is to assess to what extent this goal has been accomplished. There is one common problem which arises and it is the liaison that exists between job seeker and job portal company. The reason is that most of the recommendations do not go with to the job seekers' appropriations. Information like the location, type of job and areas of interest are received from the candidate by the normal job portal and candidates are told about the possibility of work. Due to this, the possibility, the work opportunities and even the areas are limited.

In contrary to this, location based job recommendation portal accepts only the location of the job seekers. It then provides them with various job opportunities which are at various locations with the location of the seeker centralized. Due to this system, job seekers are provided with opportunities in different locations along with the distance from the job seekers place and they are satisfied in various fields. Users are allowed to see where their friends are, to search location – tagged content within their social group and to meet others nearby through Location based social network system. LBSNs are social networking where geographic services and capabilities like geo – coding and geo – tagging are used to facilitate additional social dynamics.

Algorithms' Challenges

There are more challenges which are faced by Recommendation algorithms for the social web (*i.e*. LBSNs) than recommendation algorithms for the traditional social web. There are certain differences which exist between traditional OSNs and LBSNs. In the forth coming paragraphs, we briefly discuss each characteristic/requirement that should be possessed by recommendation algorithms for use in LBSNs.

Need for More Compact GUI : LBSNs consider that the traditional graphical user interface of OSNs is very old. An important necessity to visualize users' movements in the geo-locational dimension is the need for the presence of a map in the GUI of the LBSNs.

Node Heterogeneity : There are graphs which are derived from LBSNs and they are more complicated. They can be unipartite (*i.e*. user-user), bipartite (*i.e*. user-location) or even tripartite graphs (*i.e*. user-location-activity).To say shortly, there is a high percentage of node heterogeneity and it challenges recommendation algorithms [21].

Need for Scalability : LBSNs are developing at a faster rapidity than traditional OSNs [21]. For instance, it is common for users to visit new places without any (Table 2 OSNs vs. LBSNs) algorithmic requirements as prerequisites. In contrary to this, in an academic OSN, the mobility of users to different level conferences are not easy.

Table. 2.

Algorithms' requirements	*OSNs*	*LBSNs*
Need for compact GUI	Medium	High
Node heterogeneity	Medium	High
User's privacy protection	Medium	High
Need for scalability	Medium	High
Need for fast response	Medium	High
Need for fast response	Medium	High

Need for Fast Response : LBSN users are usually busy. They carry a device that is resource-limited (less storage and power supply). Due to this, there is a need for the recommendation results to be chosen appropriately and delivered quickly [5]. It is because users have a shorter attention span as they are typically moving from place to place.

User's Privacy Protection : It is the nature of LBSNs to impose strong privacy barriers. It is because questions for the safety of user's along several dimensionsare raised due to the imprinting of such sensitive information in the web, (*i.e*. the user's physical location) [2, 15]. For this reason, there is extensive related work regarding the user's privacy [9, 10,11].

Sparsity : This is a situation in which the users are reluctant to provide information about their current location . Due to this, data sets become sparse. The problem of sparsity is broadly studied. In such cases, precise proposals cannot be provided by the recommendation engine and it is because of the lack of sufficient information. There is another problem which is related to the new incoming users or location and it is called the "cold-start" problem. Again, the system does not have adequate information in order to endow the new registered users with recommendation or new inserted locations.

Categories of Recommendation Algorithms

There are numerous algorithms that provide user, location or activities recommendations based on mobile usage data of the users. For instance, there are systems that use collaborative filtering solutions, graph-based models, tensor-based methods, etc. We can discuss all these algorithms in two main categories as shown in figure 3.

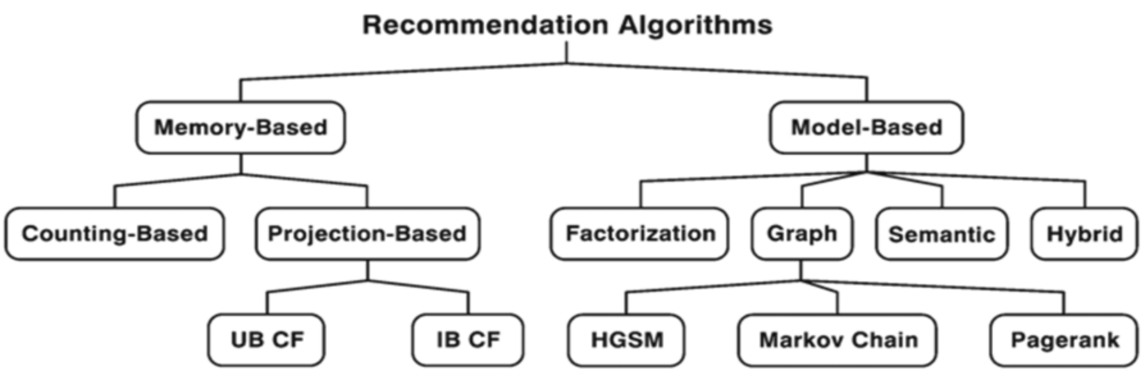

Fig. 3. Classification of recommendation algorithms.

- Memory-based algorithms perform computations directly on the complete database so that the recommended items are identified to a target user. These algorithms are inexpensive to compute and they have also proven to attain good results.
- Model-based algorithms recommend items to a target user after their have first computation of a model. These algorithms have the operating cost to build and update the model. Once the model has been built by them, good scalability is presented by them.

EVALUATION

Metrics

Some basic metrics for evaluating the effectiveness of recommender systems are mentioned. We have noticed the Mean Absolute Error (MAE), the receiver-operating characteristic ROC, and Precision-Recall as well and they mainly start off from the Information Retrieval field. When a fairer metric is compared with Precision-Recall then it is Mean Average Precision (MAP). It takes the order of a hit in the recommendation list as well into consideration. That is, since a Framework test user *u* is provided with a top-*k* list of friends/location/activities/ events, it is significant to consider the order of the presented friends/locations/activities/events in this list. Let's imagine that we recommend friends to a user. Then, it is better to have a correct guess in the first places of the recommendation list. Thus, we use the *Mean Average Precision*

(MAP) to emphasize ranking of relevant users higher. We define MAP by Eq

$$\text{MAP} = \frac{1}{[\text{N}]}\sum_{u=1}^{|\text{N}|}\frac{1}{r_u}\sum_{k=1}^{r_u} \text{Precison}_u\ @k$$

where N is the number of users in the test data set, *ru* is the number of relevant users to a user *u* and Precision *u@k* is the precision value at the *k*-th position in the recommendation list for *u*. We have to note that MAP takes both precision and recall into account and is geometrically referred to as the area under the Precision-Recall curve. W must also note that there is another metric similar to MAP and it is *n*DCG (Normalized Discounted Cumulated Gain).

There is an equivalent metric to the receiver-operating characteristic (ROC) curve and it is the area under the ROC curve known as AUC statistic and it quantifies the accuracy of prediction algorithms and also estimates the improvement over pure chance. The possibility is that a correctly chosen existent friend/location has a higher similarity value than a randomly chosen non-existent friend/location. In the implementation among n times of independent comparisons, if there are n0 times the correct prediction of friend/location has higher similarity value n00 times the correct predicted friend/location and nonexistent friend/location have the same similarity value, then we define AUC by Eq.

$$\text{AUC} = \frac{n' + 0.55xn''}{n}$$

If all similarity values are generated from an independent and identical distribution, then the accuracy should be about 0.5. Therefore, the degree to which the accuracy goes beyond 0.5 specifies the occurrence of the algorithm over pure randomness.

4. GENERIC RECOMMENDATIONS

CLAF Algorithm

Zheng et al. [21] proposed CLAF, which provides generic location recommendations within Geo life [20], a prototype LBSN proposed by Microsoft Research Asia.1 In particular, they propose activities to users for a given location and locations for a given activity. CLAF makes use of the relations between activities in its model. For example, when a user goes to a cinema, then she may also go to a restaurant,. This is called a kind of latent relationship and it is intended to be captured as activity correlation.

What should be noticed is that CLAF includes information from both the features of the locations and the activity correlations. As shown in Fig. 4, additional information captured in Location-Feature and Activity-Activity matrices is used by CLAF so that missing entries in the Location-Activity matrix may be predicted and it is generally very meager. CLAF is alike Collective Matrix Factorization, originally proposed by Singh and Gordon [16]. In the Collective Matrix Factorization model, an objective function is converted to an optimization problem and it is later solved iteratively by *Gradient Descent*.

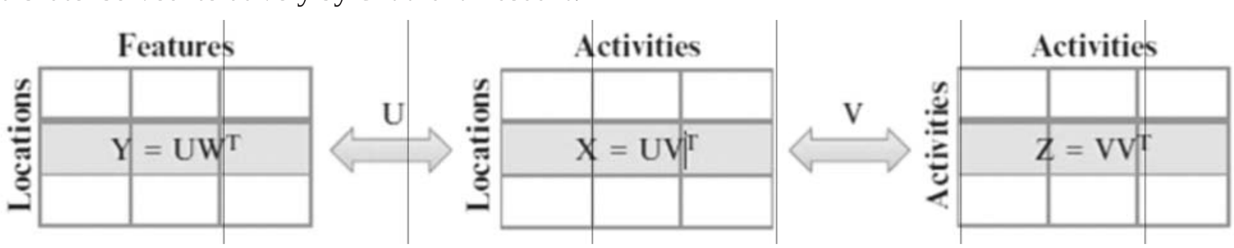

Fig. 4. Location and activity measures.

IFC Algorithm

Sattari et al [13] proposed an extended matrix model called Improved Feature Combination (IFC), and it assimilates data from one resource (e.g. Location-Activity matrix) with data from additional resources (e.g. Activity-Activity and Location-Feature matrices) to leverage generic Location/Activity recommendations. As shown in Fig. 5, preference ratings of users on a specific activity in a specific location are implied in the Location-Activity sub-matrix. In fact, its entries correspond to the frequency of performing an activity in that location for all users. The Location-Activity sub-matrix contains available features of locations and the Activity. Activity sub-matrix symbolizes relationships (semantic or others) among diverse activities. A model is built by IFC and additional information is injected into the main data by building an extended matrix. Then, Singular Value Decomposition (SVD) is applied on the merged data in order to extract hidden relations between locations, their features and activities that users perform. Numerous experiments have been conducted in which the results of IFC were compared with CLAF [21]. It was shown in their experiments that IFC outperforms CLAF in terms of prediction accuracy.

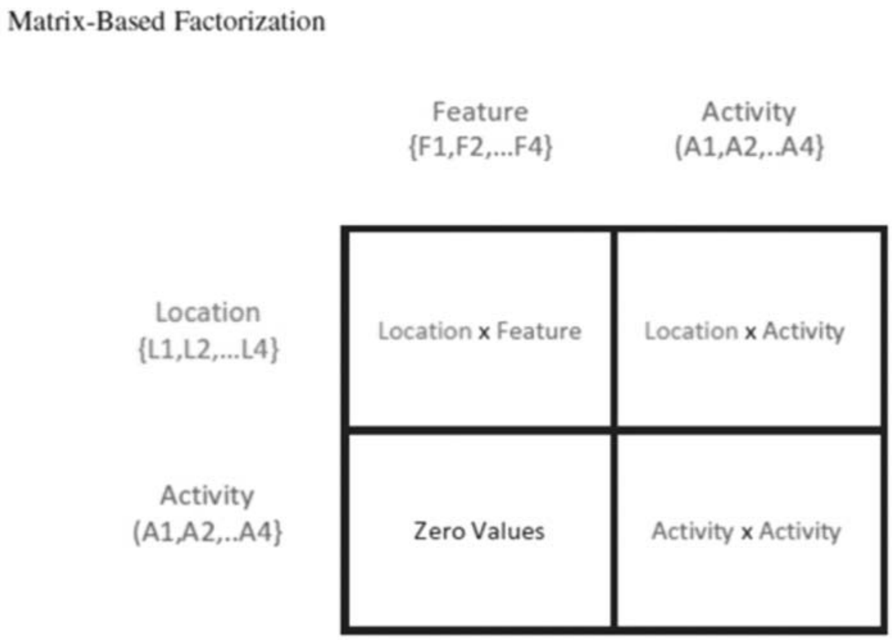

Fig. 5. Rating of user based on activity and location

PERSONALIZATED RECOMMENDATIONS

PCLAF Algorithm

Zheng et al [19] brought in a personalized recommendation algorithm for LBSNs and it performs Personalized Collaborative Location and Activity Filtering (PCLAF). Contrary to CLAF [21], PCLAF treats each user in a different way and uses a collective tensor and matrix factorization so that personalized recommendations are provided. The uniqueness of PCLAF lays on the utilization of a User-Location-Activity tensor along with the User-User, User-Location, Location- Features and Activity-Activity matrices. In order to fill missing entries in the tensor *A*, PCLAF decomposes *A* w.r.t. each tensor dimension (i.e. user, location, activity). Then, the latent factors are forced to be shared with the additional matrices by PCLAF to utilize their information. After obtaining such latent factors, an approximation tensor *AO* is reconstructed by PCLAF by filling all the missing entries. What should be noted is that PCLAF uses a PARAFAC-style regularized tensor decomposition framework so that the tensor is integrated with the additional matrices.

CADC Algorithm

A part of the Collaborative Location Recommendation (CLR) framework is the CADC algorithm [6]. CLR collects GPS trajectory data of the users and it is represented with a graph-based structure. It is denoted as Community Location Model or CLM graph. The Community-based Agglomerative-Divisive Clustering (CADC) algorithm co-clusters the data of the CLM graph into groups of alike users, alike locations and alike activities. Besides, incremental updates of the groups are supported by CADC when there is new arrival of GPS trajectory data. To say in short, the new inserted data are incrementally re-clustered without re-clustering the whole CLM graph. Different user classes are considered by CLR (*i.e*. pattern users, normal users and travelers) in accordance with the sequence of locations that they visited in their daily activities. Thus, location recommendations from different classes of users can be provided by CLR. When we take the nature of CLM graph into consideration, it is a tripartite graph and it effectively captures the relations between users, activities and locations (refer figure 6).

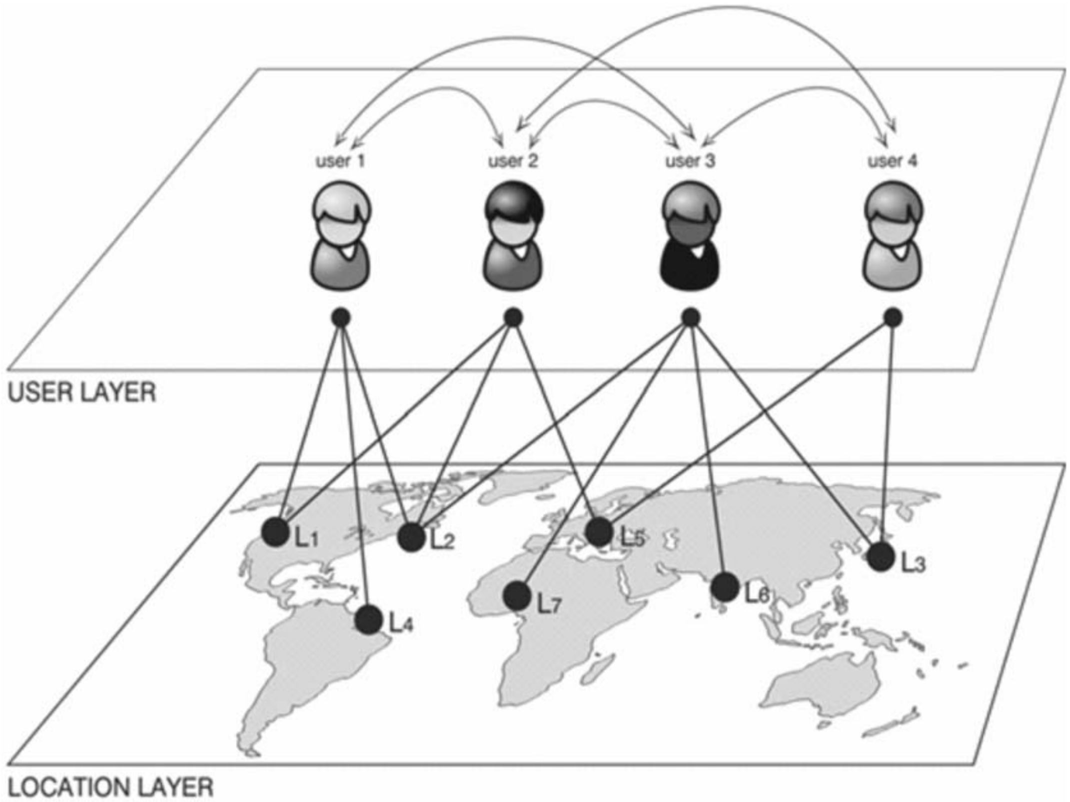

Fig. 6. Two layer representation of user and location.

Let us imagine that the User-Location model is a bipartite graph, whereas the User-Activity and Activity-Location Model are two separate bipartite graphs. Contrary to the aforementioned models, the CLM model uses: (1) the spatial properties of locations, (2) the temporal-spatial properties of activities, as they are temporal sequences of visited locations, and (3) the long-term spatial properties of users, as they visit different locations due to their long term habits or geographical limits.

Recommendation Type

We observe that Table 3 shows the categorization of algorithms in terms of recommendations types (i.e. friend, location, activity, and event). As shown, it is focused by most methods on providing either location or activity recommendations. But, many methods cannot provide more than two types of recommendations. Specifically, the only method that provides three different types of recommendations is ITR. Eventually, a specialized category of activity recommendation, *i.e.*, event recommendation, deserves further research attention.

Table 3.1 : Algorithms vs. generic/personalized recommendations.

Algorithm	*Generic*	*Personalized*
CLAF	□	
IFC	□	
PCLAF		□
CADC		□

Table 3.2 : Algorithms vs. recommendation types

Algorithm	*Location*	*Activity*
CLAF	□	□
IFC	□	□
PCLAF	□	□
CADC	□	

Table 3.3 : Data Categories of Problem Modeling.

Algorithm	*Matrix*	*Tensor*	*Graph*
CLAF	□		
IFC	□		
PCLAF		□	
CADC			□

Table 3.4 : Categories of recommendation algorithms.

Algorithm	*Collab filtering*	*Factorization*	*Clustering*
CLAF	□	□	
IFC	□	□	
PCLAF			
CADC			□

5. CONCLUSION

The main focus of this paper is on fundamental information for algorithms in Recommender Systems. In particular, basic algorithms and metrics which are used in the field of Recommender Systems and are specially used in Location Based Recommender System (LBRS) are considered. The basic concepts and recommendation algorithms proposed for Online Social Networks (OSNs) are dealt with. Recommendation algorithms for the Location-based Social Networks (LBSNs) are also formed. A few challenges of recommendation algorithms in LBSNs and the main algorithmic families are studied in this paper. We can conclude from this study that most of these algorithms are designed in such a way so as to recommend the location on the basis of the user rating values and frequency of the user utilization. However, these algorithms should be improved so that job portals can be implemented for effective recommendations of job vacancies.

6. REFERENCES

1. Alexis Papadimitriou, Panagiotis Symeonidis, Yannis Manolopoulos, “A generalized taxonomy of explanations styles for traditional and social recommender systems”, *Springer*, 2011.
2. Associated Press, Stalker victims should check for GPS. Technical Report, CBSNews.com, Feb 2003
3. J. Breese, D. Heckerman, C. Kadie, Empirical analysis of predictive algorithms for collaborative filtering, in *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence (UAI)*, Madison, WI (1998), pp. 43–52
4. B. Mobasher, H. Dai, T. Luo, M. Nakagawa, Improving the effectiveness of collaborative Filtering on anonymous web usage data, in *Proceedings of the IJCAI Workshop on Intelligent Techniques for Web Personalization (ITWP)*, Seattle, WA (2001), pp. 53–60
5. T. Horozov, N. Narasimhan, V. Vasudevan, Using location for personalized POI recommendations in mobile environments, in *Proceedings of the International Symposium on Applications on Internet (SAINT)*, Washington, DC (2006), pp. 124–129

6. N. Li, G. Chen, Analysis of a location-based social network, in *Proceedings of the International Conference on Computational Science and Engineering (CSE'2009)*, vol. 4, Vancouver (2009), pp. 263–270

7. K.W.T. Leung, D.L. Lee,W.C. Lee, CLR: a collaborative location recommendation frame work based on co-clustering, in *Proceedings of the 34th ACM SIGIR International Conference on Research and Development in Information Retrieval (SIGIR)*, Beijing (2011), pp. 305–314

8. Linyuan Lu, Matus Meod, Chi Ho Yeung, Y-cheng Zhang,Zi-Ke Zhang, Tao Zhou, "Recommender Systems", *Physics Reports 519* (2012), 1 – 49.

9. B. Lee, J. Oh, H. Yu, J. Kim, Protecting location privacy using location semantics, in *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, San Diego, CA (2011), pp. 1289–1297

10. K. Puttaswamy, N. Zhao, Preserving privacy in location-based mobile social applications, in *Proceedings of the 11th Workshop on Mobile Computing Systems and Applications (HotMobile)*, Annapolis, MD (2010), pp. 1–6

11. D. Quercia, S. Hailes, Sybil attacks against mobile users: friends and foes to the rescue, in *Proceedings of the 29th Conference on Information Communications (INFOCOM),* San Diego, CA (2010), pp. 336–340

12. D. Saez-Trumper, D. Quercia, J. Crowcroft, Ads and the city: considering geographic distance goes a long way, in *Proceedings of the 6th ACM Conference on Recommender Systems (RecSys)*, Dublin (2012), pp. 187–194

13. B. Sarwar, G. Karypis, J. Konstan, J. Riedl, Application of dimensionality reduction in recommender system - a case study, in *Proceedings of the ACM SIGKDD Workshop on Web Mining for E-Commerce - Challenges and Opportunities (WEBKDD)*, Boston, MA (2000)

14. M. Sattari, M. Manguoglu, I.H. Toroslu, P. Symeonidis, P. Senkul, Y. Manolopoulos, Geo activity recommendations by using improved feature combination, in *Proceedings of the ACM Ubi Comp International Workshop on Location-Based Social Networks (LBSN)*, Pittsburgh, PA (2012), pp 996–1003

15. B. Schilit, J. Hong, M. Gruteser, Wireless location privacy protection. IEEE Comput. **36**(12), 135–137 (2003)

16. A.P. Singh, G.J. Gordon, Relational learning via collective matrix factorization, in *Proceeding of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, Las Vegas, NV (2008), pp. 650–658

17. Dr.K.Thangadurai,M.Venkatesan, *Impact on Social Filtering and Meta Filtering In Recommender Systems Computing and Communication Technologies (WCCCT), 2014 World Congress on Feb. 27 2014-March 1 2014, PP*

18. M. Ye, P. Yin, W.C. Lee, D.L. Lee, Exploiting geographical influence for collaborative point of- interest recommendation, in *Proceedings of the 34th ACM SIGIR International Conference on Research and Development in Information Retrieval (SIGIR),* Beijing (2011), pp. 325–334

19. V. Zheng, B. Cao, Y. Zheng, X. Xie, Q. Yang, Collaborative filtering meets mobile Recommendation: a user-centered approach, in Proceedings *of the 24th AAAI Conference on Artificial Intelligence (AAAI)*, Atlanta, GA (2010)

20. Y. Zheng, X. Xiem, W.Y. Ma, Geolife: a collaborative social networking service among user, location and trajectory. IEEE Data Eng. Bull. **33**(2), 32–39 (2010)

21. V. Zheng, Y. Zheng, X. Xie, Q. Yang, Collaborative location and activity recommendations with GPS history data, in *Proceedings of the 19th International Conference on World Wide Web (WWW)*, New York, NY (2010), pp. 1029–1038

22. Y. Zheng, X. Zhou, *Computing with Spatial Trajectories* (Springer, Berlin, 2011)