Orchestration of POIs Ubiquitous Contexts: a Review of Recommendation Systems Based on Matrix Factorisation Model

Hadjhenni Mhamed, Dennouni Nassim and Slama Zohra

December 21, 2023

Orchestration of POIs Ubiquitous Contexts: A Review Of Recommendation Systems Based On Matrix Factorisation Model

Mhamed HADJHENNI
EEDIS Laboratory
Djilali Liabes University
Sidi Bel Abbes, Algeria
md.hadjhenni@univ-chlef.dz

Nassim DENNOUNI
LIA laboratory
Hassiba Benbouali University
Chlef, Algeria
n.dennouni@univ-chlef.dz

Zohra SLAMA
EEDIS Laboratory
Djilali Liabes University
Sidi Bel Abbes, Algeria
zslama@yahoo.com

Abstract— **Currently, to take full advantage of the capabilities of Artificial Intelligence (AI), Smart tourism must use Context-Aware Recommendation Systems (CARS) to orchestrate the evolving contexts of users with smartphones in order to improve their travel experiences. This type of orchestration allows points of interest (POIs) recommendations to be personalised according to the ubiquitous context of tourists during their visits. Recommending the next POIs to visit can be based on collaborative filtering techniques founded on memory or models such as matrix factorisation (MF). This paper explains the contribution of approaches that integrate contexts into models, such as MF, compared to collaborative filtering approaches without context. Consequently, this survey shows that collaborative filtering techniques using MF considerably alleviate the problems associated with the cold start of CARS and that the three types of orchestration of tourist contexts (pre-filtering, post-filtering or context modelling) improve their satisfaction.**

Keywords— CARS, POI, orchestration, collaborative filtering, Matrix factorisation, context pre-filtering, context post-filtering, context modelling.

I. INTRODUCTION

Artificial intelligence offers several potential applications in the tourism industry. AI is a promising way for tourists to discover critical data, enabling better mobility, improved decisions and an exceptional tourism experience [1]. The rise of AI is particularly relevant to almost every type of business. In the tourism sector, AI is employed for a wide range of purposes, including boosting levels of individualisation, personalising consumer suggestions and providing fast reaction times, even when staff are unavailable. The existence of artificial intelligence in the industrial environment has become so vital that it assists and communicates with customers, thereby increasing the quality of engagement. AI techniques include recommender systems (RSs), which enable relevant information to be proposed to users from a large mass of information.

The main types of RSs are: (1) content-based RSs, which use the characteristics of items to suggest articles, products, or content similar to those the user has already enjoyed or consulted. They analyse the intrinsic attributes of items, such as keywords, categories, or themes, to offer suggestions that are personalised and relevant to the user's preferences [2].

(2) Recommendation systems based on collaborative filtering identify a user's preferences based on the behaviour or evaluations of other similar users using data collected on past interactions. These systems recommend items by predicting the user's interests based on the tastes or actions of other individuals sharing similarities in their choices [3].

The combination of the two previous categories is known as (3) hybrid RSs, which makes it possible to benefit from the advantages of both categories [4].

RSs can be used in several domains, including tourism, for travel recommendations of the next POI to visit. RSs problems include the cold-start problem that occurs when a new user or item is added [5]; in this case, the system cannot generate a personalised recommendation due to a lack of information. RSs based on MF can solve this problem; in this article, we will study RS based on MF and see their contribution to the POI recommendation domain. Another problem with RSs is that users are not satisfied with the recommendations they receive, mainly because RSs need to consider the user's context. Integrating context into RSs can improve user satisfaction with the recommendations they receive.

Context is any information that can be used to characterise the situation of an entity. An entity can be a person, place or object considered relevant to the interaction between a user and an application, including the user and the application itself [6]. In specific domains, RSs use contextual information such as time, location, and companion to improve the recommendation process. This type of system is called a Context-Aware Recommendation System (CARS) [7] [8]. This contextual approach improves the accuracy of recommendations by adapting those to specific situations, offering more relevant and engaging user experiences. It also provides a better understanding of users' changing needs, improving satisfaction and loyalty by offering better-targeted recommendations tailored to the user's context.

In this article, we surveyed works related to CARSs based on collaborative filtering, particularly those based on model and matrix factorisation techniques. We have discussed and compared work on FM and context integration to propose future directions for this type of technique in tourism.

This paper is structured as follows. Section II reviews the relationship between orchestration, recommender systems and context for the tourism domain. After that, in Section III, we present work related to the definition of context and approaches to incorporate context within collaborative filtering-based recommender systems using matrix factorisation. Section IV will be devoted to comparing different collaborative filtering-based RSs (memory-based and matrix factorisation-based). In section V, we discuss the use of these techniques for the POI recommendation domain. Finally, in section VI, we summarise the contributions of our paper in order to propose future perspectives for our work.

II. PRELIMINARY NOTIONS

In this section, we look at the principle of POI orchestration using RS/CARS and the personalisation of POI recommendations based on the integration of user contexts.

A. *RS and CARS*

Most RS approaches focus on recommending the most relevant items to users and ignore contextual information, such as the time, place and companionship of other people (e.g. watching movies or dining out with friends or family). For this reason, traditional RSs deal with only two types of entities, users and items, and do not consider context in the recommendation process. However, it is essential to integrate contextual information into the recommendation process to suggest items to users in many applications, such as a travel recommendation application. For example, by using the context of the year's season, the travel RS can suggest skiing in winter and the beach in summer.

In the literature, three approaches can be used with CARS, depending on when the context was injected [9]:

Contextual pre-filtering consists of selecting a subset of data for the context in which it is found and limiting the recommendation process to this subset. The RS builds a model for each context [10]. For example, if a user wants to take a trip at the weekend, only POIs that are open at the weekend can be recommended, and only the scores of users who visited POIs over the weekend are used to predict the score. This pre-filtering can cause score prediction problems if the system does not have enough data (data sparsity problem).

In ***contextual post-filtering,*** the RS does not consider contextual data during the recommendation process. The results of the recommendation algorithms are reorganised according to the context to produce the list of items to be recommended [10]. For example, a system that recommends touristic places will use the user's geolocation (location context) and may decide to omit subsequent recommendations for places that are too far from the user's location.

Finally, the ***contextual modelling*** approach incorporates contextual information directly into the recommendation process to predict item scores. A recommendation is no longer considered a function with parameters such as items and users but a function described with items, users and context variables [10].

B. *Orchestration and RS*

Orchestrating tourist visits involves planning and coordinating various aspects to offer tourists a harmonious experience. Using RS/CARS involves recommendation algorithms to personalise itineraries according to tourists' preferences.

RSs based on Collaborative Filtering (CF) can be divided into two main categories: memory-based and model-based collaborative filtering [3]. Memory-based collaborative filtering exploits similarity between users or items, while model-based collaborative filtering uses mathematical modelling or machine learning techniques (matrix factorization or neural networks) to predict user preferences.

C. *Orchestration, context and CARS*

Integrating the ubiquitous context in tourism relies on using technologies such as GPS to adjust real-time recommendations according to location, weather and other factors. Various approaches facilitate the orchestration of contexts for mobile tourists, using sensors, mobile applications, wearable devices and augmented reality. These systems collect location, time, weather conditions and tourist behaviour data, offering personalised recommendations with CARS. Visit orchestration encompasses trip planning, post-arrival navigation, content personalisation, and integrated customer service to resolve any problems encountered by tourists effectively. The aim is to deliver a seamless, personalised travel experience adapted to ubiquitous contexts, representing a key area of research and development to improve overall tourist satisfaction.

III. LITERATURE REVIEW

Matrix factorization techniques use different models, including Matrix factorization (MF) [11], Sparse Linear Method (SLIM) [12], Singular Value Decomposition (SVD) [13], Probabilistic Matrix Factorization (PMF) [14] and Non-negative Matrix Factorization (NMF) [15]. In This section, we will explain the operating principles of FM-based RSs AND SLIM RS.

A. *RS FM-based functioning*

The recommendation system based on matrix factorisation decomposes a matrix of user-item ratings into two smaller matrices: a user matrix and an item matrix, which capture the underlying relationships between users and items.

In MF [11], users and items are represented by vectors p and q, respectively. The values of these vectors indicate the weights of K (e.g., K = 5) latent factors. Therefore, the ranking prediction is described by Equation 1.

$$\hat{r}_{ui} = \overrightarrow{P_u} . \overrightarrow{q_i} \qquad (1)$$

The user and item biases rating (deviations) can be added to the prediction function, as shown in equation 2, where μ represents the overall average rating in the dataset, b_u and b_i represent the user and item rating biases, respectively [16].

$$\hat{r}_{ui} = \mu + b_u + b_i + \overrightarrow{P_u} . \overrightarrow{q_i} \qquad (2)$$

Loss Function:

To train the model, we need to define a loss function that measures the difference between the predicted values and the actual values of the R matrix. The root mean square error (RMSE) is one of the most commonly used loss functions.

$$Loss = \sum (R_{ij} - \hat{R}_{ij})^2 \qquad (3)$$

The matrix factorisation can be done by using optimisation techniques, such as stochastic gradient descent, to adjust the values of U and V to minimise the loss function. From that, we can learn the latent user and item characteristics. This training process is iterative, and the model converges to values of U and V that best minimise the difference between the predicted and actual values of the R matrix. Once the model has been trained, personalised recommendations can be generated by calculating predictions for items not rated by a user and ranking them according to these predictions.

Matrix factorisation can also be applied to the tourism sector to recommend Points of Interest (POIs) and discover interesting places. Matrix factorisation for user-POI (Point of Interest) recommendations are based on the decomposition of a user-POI interaction matrix into two smaller matrices: a user matrix (U) and a POI matrix (V).

Here is a simplified example of matrix factorisation with some detailed formulae involved. Suppose we have a user-POI interaction matrix (R) of dimensions M x N, where M represents the number of users and N is the number of POIs. Each element of the matrix R (R_{ij}) represents the interaction of $user_i$ with POI_j, for example, a rating given by the user to a POI.

The aim is to decompose this matrix R into two matrices, U and V, so their product approximates the matrix R as closely as possible. This process models user preferences and POI characteristics as vectors.

User matrix (U)

The matrix U has dimensions M x K, where K is the number of latent features (factors). Each row of the matrix U represents a user, and each column represents a latent characteristic. The value U_{ik} represents the extent to which user i is influenced by latent feature k.

POI matrix (V)

The matrix V has dimensions K x N. Each row of the matrix V represents a latent feature, and each column represents a POI. The value V_{kj} represents the extent to which latent feature k is present in POI j.

Predicting interactions

The prediction of the interaction (score) of a user i with a POI j is obtained by the scalar product between the user vector Ui and the POI vector Vj and the addition of a global bias (or user and POI biases):

$$\boldsymbol{R_{ij} = U_i * V_j + G\_biais + U\_biais_i + P\_biais_j} \quad \textit{(4)}$$

The biases $\mathbf{G\_biais}$, $\mathbf{U\_biais_i}$ **and** $\mathbf{P\_biais_j}$ (global bias, user bias and POI bias) are terms added to account for global effects and the particularities of each user and POI.

Using the following example, we will explain how a model-based recommendation system works using matrix factorisation. Consider the following matrix of ratings given by users U1, ..., U5 to POI1,...POI5.

U/P	POI 1	POI2	POI3	POI4	POI5
U1	?	5	4	2	1
U2	1	?	?	5	3
U3	1	4	4	1	?
U4	?	?	2	?	2
U5	3	1	1	?	?

For this matrix R, we will define the number of latent factors with the value K=2 and carry out several iterations to obtain the matrix factorization of R.

First iteration:

We randomly initialise the matrix P (U, K) and Q(K, I). In our case, the matrices P(U,K) and Q(K,I) are defined as follows:

P(5*2)

	K1	K2
U1	0.16	0.25
U2	0.67	0.26
U3	0.72	0.07
U4	0.81	0.05
U5	0.96	0.83

Q (2*5)

	POI 1	POI 2	POI 3	POI 4	POI 5
K1	0.49	0.86	0.52	0.03	0.69
K2	0.58	0.15	0.82	0.50	0.46

$\boldsymbol{\hat{R}}$ =

	POI 1	POI 2	POI 3	POI4	POI5
U1	0.22	0.17	0.28	0.13	0.22
U2	0.48	0.61	0.56	0.15	0.58
U3	0.39	0.63	0.43	0.06	0.53
U4	0.43	0.71	0.47	0.05	0.58
U5	0.95	0.95	1.18	0.44	1.04

N iterations After N=100000, we will have:

P (5*2)

	K1	K2
U1	1.93	0.75
U2	-1.30	1.75
U3	1.74	0.37
U4	0.79	1.27
U5	0.14	2.51

Q (2*5)

	POI 1	POI 2	POI 3	POI 4	POI 5
K1	0.46	2.33	2.05	-0.05	-0.11
K2	1.07	0.28	0.28	2.75	1.60

$\boldsymbol{\hat{R}}$ =

U/P	POI 1	POI 2	POI 3	POI4	POI5
U1	1.69	4.72	4.17	1.98	0.99
U2	1.27	-2.55	-2.18	4.87	2.95
U3	1.19	4.16	3.67	0.93	0.40
U4	1.72	2.20	1.97	3.46	1.95
U5	2.75	1.05	0.99	6.91	4.01

B. RS SLIM-based functioning

SLIM is a model-based collaborative filtering method that aims to solve the Top-N recommendation problem. SLIM is based on a sparse matrix factorisation approach, emphasising smoothing to obtain a sparse matrix that models the relationships between items. This approach produces high-quality recommendations while being efficient in terms of computational complexity. It is derived from the FM method, but instead of looking for $\boldsymbol{\hat{R} \approx P.Q}$, we look for $\boldsymbol{\hat{R} \approx RQ}$ in the SLIM method. Q can represent the deviation matrix between items (POIs), the SLIM I variant, with K= number of items (POIs). Alternatively, Q can represent the deviation matrix between users, and this is the SLIM U variant, with K= number of users.

FM / SLIM comparison

- FM focuses on modelling the latent characteristics of users and items, while SLIM focuses on modelling the similarity between items or users.
- FM is best suited for personalised recommendations based on latent characteristics, whereas SLIM can recommend items similar to those the user has previously enjoyed.
- FM may be more appropriate for available explicit data, such as ratings, as it can predict accurate ratings.
- SLIM may be more appropriate when diversifying recommendations based on item similarities.
- It is also possible to integrate these two approaches to obtain a combination of the advantages of both methods.

C. *CARS FM/SLIM-based functioning*

Based on the variants for integrating context into a recommendation system, we can have the following CARS alternatives with FM and SLIM.

FM/SLIM with Pre-filtering

This type of system combines the advantages of model-based collaborative filtering with context-dependent customisation. Adding context in pre-filtering means contextual data, such as location or time of day, are considered before the recommendation process. This approach adapts suggestions according to the specific context in which the user interacts with the system.

FM/SLIM with post-filtering

FM /SLIM RS incorporates context after the initial generation of recommendations. This approach adjusts suggestions according to the user's specific context, and the system becomes more flexible in adapting to the evolving preferences of users in different situations.

FM/SLIM with Context Modelling

Using context modelling with matrix factorisation allows the system to integrate dynamically contextual data into modelling relationships between users and items. This approach enables real-time adaptation to context variations, improving recommendations' relevance.

IV. Comparison of FM-based approaches

In the literature, in addition to the FM and SLIM methods, there are other variants of the FM method for RSs, such as:

(1) The **SVD** method decomposes a matrix R into three matrices, capturing the most significant components and truncating them to reduce dimensionality while preserving important information. SVD factors a matrix into its main components, representing users and items in a space of reduced dimensions. This method, used in recommender systems, results in the original matrix R decomposed as follows: $\mathbf{R = U{*}\Sigma{*}VT.}$

(2) The **PFA** method is often used when the data is not simply continuous but follows probabilistic distributions. It is used to model user-item interactions using probabilistic concepts. PMF aims to approximate a user-item matrix R by two feature matrices, P for users and Q for items, by maximising the probability of observing existing evaluations.

(3) The **NMF** method is similar to matrix factorisation but restricts the resulting matrices to positive values, allowing a more straightforward interpretation of the components. It can be seen as a variant of matrix factorisation but with additional constraints on component values.

In order to be able to compare the above approaches with each other and with collaborative filtering techniques, we have set the following criteria:

a) Cold start

The cold-start problem in recommender systems occurs when new users or items without a significant history are introduced; it compromises the accuracy of similarity-based collaborative filtering.

b) Data sparsity

Data sparsity concerns the scores available in user-item matrices, where most users have only rated a fraction of the available items. It creates a crucial challenge: predicting preferences for unrated items. It complicates the task of recommendation algorithms, as they have to infer relationships between users and items based on incomplete data. Approaches to solving this problem include the use of matrix completion techniques and matrix factorisation models to improve the quality of recommendations despite sparse data.

c) Context integration (temporal, geographical, seasonal)

Context integration, whether temporal, geographical, or seasonal, is crucial when evaluating recommender systems. Exploring and using different contexts contributes to an in-depth understanding of user preferences, ensuring personalised and relevant recommendations.

d) Prediction significance

The significance of predictions in a recommendation system evaluates the system's ability to provide understandable explanations behind the suggestions it generates. This criterion encompasses the model's explicability, the decision-making process's transparency, consideration of the specific characteristics of an item or a user's preferences, and the possibility of the user having control over the recommendation process.

A straightforward interpretation of predictions increases user confidence and facilitates a more informed and satisfying interaction. By focusing on the comprehensibility of the system, the meaning of predictions promotes a transparent and adaptable user experience.

e) Recommendation personalisation

Personalising recommendations involves adapting suggestions according to each user's preferences. The aim is to create a unique user experience, improving the relevance of suggestions and user satisfaction.

f) The diversity of recommendations

Diversity refers to a RS ability to offer various suggestions rather than focusing on similar choices. The aim is to avoid redundancy by presenting users with diverse items or content, reflecting different aspects of their preferences, to ensure a richer user experience.

Table I. Comparison of Collaborative Filtering Approaches

Criteria	MBCF[1]	FM[2]	SLIM[3]	SVD[4]	PFA[5]	NMF[6]
Solving the cold start problem	No	Yes	Yes	Yes	Yes	Yes
Solving the data sparsity problem	No	Yes	Yes	Yes	Yes	Yes
Context integration (temporal, geographical, etc.)	Yes	Yes	Yes	Yes	Yes	Yes
Adaptability to real-time changes in context	No	Yes	Yes	Yes	Yes	Yes
Prediction significance	Yes	No	No	No	No	No
Personalised recommendations	Yes	Yes	Yes	Yes	Yes	Yes
Diversity of recommendations taken into account	Yes	Yes	Yes	Yes	Yes	Yes
Scalability	No	Yes	Yes	Yes	Yes	Yes
Implementation complexity	No	Yes	Yes	Yes	Yes	Yes
Context integration : Pre-filtering, Post-filtering	Yes	Yes	Yes	Yes	Yes	Yes
Context Modelling	Yes	Yes	Yes	Yes	Yes	Yes

g) Scalability

A scalable recommender system must handle massive datasets efficiently without compromising the speed and quality of recommendations. It often involves parallel computing techniques, task distribution, or optimised data structures. Scalability is essential to guarantee a fluid and responsive user experience, even in environments with diverse users and items.

h) Implementation complexity

Higher implementation complexity can lead to higher costs and longer development times. However, moderate complexity can be justified if it enables higher performance or advanced functionality.

V. Discussion

Techniques involving matrix decomposition are based on matrix factorisation. From Table I, we can see the advantage of FM-based approaches over memory-based recommender systems. The integration of context (in pre-filtering or post-filtering) significantly contributes to improving the recommendation quality and the integration of context in the modelling context, which can dynamically adapt the recommendation results to the user's current context.

Our comparison of memory-based and FM-based approaches to recommender systems (see Table I) shows that memory-based approaches do not solve the cold-start and sparsity problems. On the contrary, the FM-based variants of RS can solve these problems. The significance of the predictions or the explanation of the recommendation results obtained is guaranteed by memory-based approaches. On the other hand, this criterion is not guaranteed by approaches based on matrix factorisation since they use latent factors whose meaning cannot be determined.

Memory-based approaches do not guarantee scalability; on the contrary, this criterion is guaranteed by matrix factorisation-based approaches but with the disadvantage of implementation complexity. All the approaches to recommender systems with collaborative filtering (memory-based and FM-based) allow for the integration of context (in pre-filtering, post-filtering, and context modelling). Results from [17][11][18] show that SVD-based prediction algorithms can effectively address the challenge of sparse data by exploiting hidden correlations. SVD performance can be improved by incorporating user and feature biases. This technique is best suited to predicting tourist check-ins. On the other hand, the FM and SLIM methods are well suited to feedback in rating or check-in format.

In the literature, there are limited dedicated FM-based RSs for POI recommendation. We have: **(1) GeoMF**: Joint Geographical Modelling and Matrix Factorization for Point-of-Interest Recommendation [19], **(2) GEOMFTD**: GEOMF with Time Dependencies POI Recommendation [20], **(3) ReGS**: Review geographical Social [21], **(4) FSS-FM:** Feature-Space Separated Factorization Model [22] and **(5) LGLMF:** Local Geographical based Logistic Matrix Factorisation Model for POI Recommendation [23]. These systems use the matrix factorisation technique and integrated context variables based on the context modelling principle to improve the recommendation's quality and guarantee tourist satisfaction. These systems use datasets such as foursquare, Gowalla, or JiePang (a Chinese location-based social

[1] Memory-Based Collaborative Filtering
[2] Factorization matricielle
[3] Sparse Linear Methods
[4] Singular Value Decomposition
[5] Probabilistic Factor Analysis
[6] Non-negative Matrix Factorization

network) to make recommendations based on feedback such as tourist check-ins. In the literature, few approaches adopt the principle of matrix factorisation, which considers several context variables such as the tourist's location and the time of the visit. For these context variables, it is necessary to consider the tourist's companion and the weather on the day of the visit. For these context variables, we propose to perform CARS based on matrix factorisation with implicit and explicit data (ratings) from other datasets such as (yelp or JiePang). We also propose considering the ubiquitous context of tourists or POIs and orchestrating the integration of these context variables to select the variables that increase tourist satisfaction. Datasets with such context data are non-existent, so building such datasets for different regions of the world would be interesting. We also propose to develop CARS based on pre-filtering and post-filtering. The evaluations and comparisons of these CARS help us to determine the best method of context integration, and to evaluate which type of data (explicit or implicit) gives better results. This evaluation can take into account the tourists' satisfaction and the remedy for the cold start problem and the sparsity problem.

VI. CONCLUSION

In this article, we present the state of the art of collaborative filtering-based RSs using matrix factorisation as a model. We then explain how FM-based approaches can alleviate the cold-start problem associated with data sparsity. We also explored how context can be integrated into the CARS FM approach to improve tourist satisfaction. This type of integration can be associated with orchestrating the different context variables using pre-filtering, post-filtering, and context modelling to produce CARS for POI. As perspectives related to our work, we wish to make online evaluations of the orchestration of context variables in FM-based RS to estimate tourist satisfaction with metrics such as Click Through (CTR) or others. This evaluation requires the implementation of several variants of the FM approach integrating context and using several datasets such as YELP or JiePang.

ACKNOWLEDGMENT

This work has been supported by the PRFU project N° COOLO7UN020120220001 entitled “Formalisation of point-of-interest recommendation systems and performance evaluation for mobile users,” Department of Computer Science, Hassiba Ben Bouali University, Chlef, Algeria.

REFERENCES

[1] Tussyadiah, I., & Miller, G. (2019). Perceived impacts of artificial intelligence and responses to positive behaviour change intervention. In Information and Communication Technologies in Tourism 2019: Proceedings of the International Conference in Nicosia, Cyprus, 2019 (pp. 359-370). Springer International Publishing.

[2] Pazzani, M. J., & Billsus, D. (2007). Content-based recommendation systems. In The adaptive web: methods and strategies of web personalization (pp. 325-341). Berlin, Heidelberg: Springer Berlin Heidelberg.

[3] Candillier, L., Jack, K., Fessant, F., & Meyer, F. (2009). State-of-the-art recommender systems. In Collaborative and Social Information Retrieval and Access: Techniques for Improved User Modelling (pp. 1-22). IGI Global.

[4] Adomavicius, G., & Tuzhilin, A. (2005). 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.

[5] Schein, A. I., Popescul, A., Ungar, L. H., & Pennock, D. M. (2002, August). Methods and metrics for cold-start recommendations. In Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval (pp. 253-260).

[6] Abowd, G. D., Dey, A. K., Brown, P. J., Davies, N., Smith, M., & Steggles, P. (1999). Towards a better understanding of context and context-awareness. In Handheld and Ubiquitous Computing: First International Symposium, HUC’99 Karlsruhe, Germany, September 27–29, 1999 Proceedings 1 (pp. 304-307). Springer Berlin Heidelberg.

[7] Baltrunas, L. (2008, October). Exploiting contextual information in recommender systems. In Proceedings of the 2008 ACM conference on Recommender systems (pp. 295-298).

[8] Karatzoglou, A., Amatriain, X., Baltrunas, L., & Oliver, N. (2010, September). Multiverse recommendation: n-dimensional tensor factorization for context-aware collaborative filtering. In Proceedings of the fourth ACM conference on Recommender systems (pp. 79-86).

[9] Panniello, U., Tuzhilin, A., Gorgoglione, M., Palmisano, C., & Pedone, A. (2009, October). Experimental comparison of pre-vs. post-filtering approaches in context-aware recommender systems. In Proceedings of the third ACM conference on Recommender systems (pp. 265-268).

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

[11] Koren, Y., Bell, R., & Volinsky, C. (2009). Matrix factorization techniques for recommender systems. Computer, 42(8), 30-37.

[12] Ning, X., & Karypis, G. (2011, December). Slim: Sparse linear methods for top-n recommender systems. In 2011 IEEE 11th international conference on data mining (pp. 497-506). IEEE.

[13] Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. T. (2000). Application of dimensionality reduction in recommender system-a case study.

[14] Salakhutdinov, R., & Mnih, A. (2008, July). Bayesian probabilistic matrix factorization using Markov chain Monte Carlo. In Proceedings of the 25th international conference on Machine learning (pp. 880-887).

[15] Cai, D., He, X., Han, J., & Huang, T. S. (2010). Graph regularized nonnegative matrix factorization for data representation. IEEE transactions on pattern analysis and machine intelligence, 33(8), 1548-1560.

[16] Zheng, Y., Mobasher, B., & Burke, R. D. (2015, July). Incorporating Context Correlation into Context-aware Matrix Factorization. In CPCR+ ITWP@ IJCAI (p. 7).

[17] Takács, G., Pilászy, I., Németh, B., & Tikk, D. (2008, August). Investigation of various matrix factorization methods for large recommender systems. In Proceedings of the 2nd KDD Workshop on Large-Scale Recommender Systems and the Netflix Prize Competition (pp. 1-8).

[18] Rendle, S., Gantner, Z., Freudenthaler, C., & Schmidt-Thieme, L. (2011, July). Fast context-aware recommendations with factorization machines. In Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval (pp. 635-644).

[19] Lian, D., Zhao, C., Xie, X., Sun, G., Chen, E., & Rui, Y. (2014, August). GeoMF: joint geographical modelling and matrix factorization for point-of-interest recommendation. In Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 831-840).

[20] Griesner, J. B., Abdessalem, T., & Naacke, H. (2015, September). POI recommendation: Towards fused matrix factorization with geographical and temporal influences. In Proceedings of the 9th ACM Conference on Recommender Systems (pp. 301-304).

[21] Xing, S., Liu, F., Zhao, X., & Li, T. (2018). Points-of-interest recommendation based on convolution matrix factorization. Applied intelligence, 48, 2458-2469.

[22] Cai, L., Xu, J., Liu, J., & Pei, T. (2018). Integrating spatial and temporal contexts into a factorization model for POI recommendation. International Journal of Geographical Information Science, 32(3), 524-546.

[23] Rahmani, H. A., Aliannejadi, M., Ahmadian, S., Baratchi, M., Afsharchi, M., & Crestani, F. (2020). LGLMF: local geographical based logistic matrix factorization model for POI recommendation. In Information Retrieval Technology: 15th Asia Information Retrieval Societies Conference, Hong Kong, China, Proceedings 15 (pp. 66-78).