GeoDCF: Deep Collaborative Filtering with Multifaceted Contextual Information in Location-Based Social Networks

Dimitrios Rafailidis[1,2](✉) and Fabio Crestani[3]

[1] Department of Computer Science, University of Mons, Mons, Belgium
dimitrios.rafailidis@umons.ac.be
[2] Department of Data Science and Knowledge Engineering,
Maastricht University, Maastricht, The Netherlands
[3] Faculty of Informatics, Università della Svizzera italiana, Lugano, Switzerland
fabio.crestani@usi.ch

Abstract. In this study we investigate the recommendation problem with multifaceted contextual information to overcome the scarcity of users' check-in data in Location-based Social Networks. To generate accurate personalized Point-of-Interest (POI) recommendations in the presence of data scarcity, we account for both users' and POIs' contextual information such as the social influence of friends, as well as the geographical and sequential transition influence of POIs on user's check-in behavior. We first propose a multi-view learning strategy to capture the multifaceted contextual information of users and POIs along with users' check-in data. Then, we feed the learned user and POI latent vectors to a deep neural framework, to capture their non-linear correlations. Finally, we formulate the objective function of our geo-based deep collaborative filtering model (GeoDCF) as a Bayesian personalized ranking problem to focus on the top-*k* recommendation task and we learn the parameters of our model via backpropagation. Our experiments on real-world datasets confirm that GeoDCF achieves high recommendation accuracy, significantly outperforming other state-of-the-art methods. Furthermore, we confirm the influence of both users' and POIs' contextual information on our GeoDCF model. The evaluation datasets are publicly available at: http://snap.stanford.edu/data/loc-gowalla.html, https://sites.google.com/site/yangdingqi/home/foursquare-dataset.

Keywords: Point-of-interest recommendation
Deep collaborative filtering · Multifaceted contextual information
Location-Based Social Networks

1 Introduction

With the emergence of Location-based Social Networks (LBSNs) such as Yelp and Foursquare, users can search for a Point-of-Interest (POI) e.g., a restaurant or a museum to visit, and share their location with their friends by making a

M. Berlingerio et al. (Eds.): ECML PKDD 2018, LNAI 11052, pp. 709–724, 2019.
https://doi.org/10.1007/978-3-030-10928-8_42

check-in at the POI they have visited. Such implicit source of feedback provides rich information about both users and POIs that can be leveraged to study the user's movement in urban cities, as well as enhance the quality of personalised POI recommendations. Most existing POI recommendation systems apply collaborative filtering techniques to suggest relevant POIs to users based on the assumption that similar-minded users are likely to visit similar POIs [7,26]. In practice, rather than explicit feedbacks of ratings for traditional recommendation systems, binary implicit feedbacks are usually available at LBSNs in the form of check-in data [20]. Several methods have been proposed to handle the case of users' implicit feedback, such as weighted matrix factorization [6], with square or cross-entropy loss functions to either minimize the rating error or predict if an unobserved item would be preferred or not by a user. However, provided that end-users are usually interested in the top-k recommendations, such loss functions do not focus on the top-k recommendation problem. To overcome this limitation, Bayesian Personalized Ranking (BPR) strategies use a pairwise ranking loss function, considering the relative ordering of items in a ranked list [17]. The pairwise ranking criterion of the BPR model is based on the assumption that a user prefers the observed items over the unobserved ones. This idea results in a pairwise ranking loss function that tries to discriminate between a small set of observed items and a very large set of unobserved ones. Due to the imbalance between the user's observed items and unobserved ones, the BPR model uniformly samples negative examples from the set of unobserved items to reduce the training time. However, both studies [6,17] ignore the multifaceted contextual information at LBSNs for POI recommendations [11,26].

POI recommendation strategies suffer from the data scarcity problem, as the number of POIs visited by a user is usually only a small portion of all the available POIs at a LBSN [2,9]. As a consequence, the data scarcity limits the performance of the collaborative filtering strategies when generating recommendations. To handle the data scarcity problem, POI recommendation strategies exploit the multifaceted contextual information of both users and POIs, such as the social influence of friends, as well as the geographical and sequential transition influence of POIs on user's check-in behavior. In particular, although user preferences are influenced by users' social relationships, the selections of social friends do not necessarily match [13,14]. As a consequence we have to learn the impact of friends' selections on users' check-in behavior [7]. Regarding POIs' geographical influence, user preferences are based on the user mobility and the geographical distances among POIs, as most users only visit POIs within small regions [10,11,21]. In addition, two users may behave differently with respect to time. For example, one often checks in at restaurants during lunch time, while the other likes bars and often checks in at midnight. The POI recommendation task becomes even more challenging, as there is a sequential transition influence of locations on users' check-in behaviors, where a user might like visiting POIs in a specific order e.g., office→, lunch, gym→home or home→bar [1,26].

Although POI recommendation strategies exploit different contextual factors, they do capture well the non-linear correlations of users' and POIs multifaceted information [20]. Also, they not necessarily focus on the ranking performance of the POI recommendation task, such as in the studies reported in [11,21]. To overcome the shortcomings of the existing methods we propose the GeoDCF model, making the following contributions: *(C1)* To account for the fact that the multifaceted information of users and POIs can significantly boost the quality of recommendations, we first introduce a multi-view joint factorization strategy. We compute the user and POI latent vectors by co-factorizing users' check-in behavior on POIs with users' and POIs' contextual information. *(C2)* To better capture the non-linear correlations of the user and POI latent vectors, we adopt a deep learning strategy by learning the model parameters via a backpropagation algorithm. *(C3)* To focus on the ranking performance, we formulate our model as a pairwise ranking task, by placing a BPR layer at the top of our deep learning architecture. Our experiments on benchmark datasets from the real-world LBSNs of Gowalla and Foursquare show that our GeoDCF model beats other baseline strategies. In addition, we experimentally show the impact of users' and POIs' contextual information on our model.

The remainder of the paper is organized as follows, Sect. 2 reviews the related work, and in Sect. 3 we formally define our pairwise ranking problem. Section 4 details the proposed GeoDCF model, Sect. 5 presents the experimental results and Sect. 6 concludes the study.

2 Related Work

In collaborative filtering with implicit feedback, such as weighted matrix factorization [6], some missing entries are treated as negative instances (negative sampling) trying to minimize a pointwise loss function such as the square or cross-entropy loss. Liu et al. [11] model geographical influence by incorporating neighboring characteristics into weighted matrix factorization to handle the implicit feedback of users' check-in data. Lian et al. [10] present a geographical weighted matrix factorization model that integrates geographical influence by modeling users' activity regions and the influence propagation on geographical space. Instead of using a pointwise loss function, Yuan et al. [23] focus on the top-k recommendation performance, presenting a model that incorporates geographical influence, assuming that neighborhood POIs of POIs previously visited by users should be ranked higher than distant ones. In a similar spirit, RankGeoFM is a ranking-based model that first learns users' preference rankings for POIs, and then includes the geographical influence of neighboring POIs to alleviate the recommendation accuracy [9].

Apart from the geographical influence on users' check-in behaviour, Ye et al. [21] also consider users' social correlation for POI recommendation, following a friend-based collaborative filtering strategy. In particular, they produce POI recommendations based on similar friends, where the similarity between friends is calculated based on their common check-in POIs and common friends. In [24], a friend-based collaborative filtering strategy is also used to leverage friends' check-ins, where the similarity between friends is computed based on the distance of their residences. In [12], a personalised ranking framework with multiple sampling criteria is proposed, leveraging both social correlation and geographical influence on users' check-in behavior. In particular, Manotumruksa et al. [12] apply a multi-center Gaussian model and a power-law distribution method, to capture the geographical influence and social correlation respectively when performing negative sampling for the non-visited POIs. In [7] a two-step POI recommendation framework is proposed, which first learns potential locations from users' friends and then, incorporates potential locations into weighted matrix factorization. Zhang et al. [26] employ an additive Markov chain to exploit the sequential transition influence between POIs, where the sequential probability of a user visiting a POI is based on the transition probability between all the user's visited POIs and a target non-visited POI.

Accordingly, in recommendation systems deep learning strategies use either a pointwise or a pairwise ranking loss function to handle user implicit feedback and capture user data non-linear correlations. In [8,18,22], various deep learning strategies are introduced to exploit user feedback with users' and items' side information. For example, Ying et al. [22] model implicit feedback in stacked denoising autoencoders with the side information of articles, such as the title and abstract of the articles. Ding et al. [3] design a ranking model for friend recommendations. However, the studies at [3,8,9,18] do not consider any contextual information when training their deep learning models, a key factor to generate accurate POI recommendations [7,21,25]. Recently, a deep recurrent neural network is proposed to capture the sequential transition influence on users' check-in behavior [4]. Nonetheless, the users' social relations are ignored at [4]. Yang et al. [20] introduce PACE, a deep neural architecture that jointly learns the embeddings of users and POIs to predict user preferences over POIs and various context associated with users and POIs. PACE first transforms the users' and POIs' contextual relations into graphs and then, employs neural embedding for POI recommendation as a bridge between collaborative filtering and semi-supervised learning. Instead of using a pairwise ranking function, PACE defines a pointwise function to handle the case of implicit feedback during the deep neural network learning. Consequently, PACE does not focus on the ranking performance when generating top-k POI recommendations.

3 Problem Formulation

Let $\mathcal{N}$ and $\mathcal{M}$ be the sets of users and POIs, where $n = |\mathcal{N}|$ and $m = |\mathcal{M}|$ are the numbers of users and POIs, respectively. Users' check-in data are tuples in the form of $(user, POI, time)$. In addition, each user u has a set of friends $\mathcal{A}_u$. Each POI is also associated with a pair of geographical latitude and longitude coordinates in the form of $(lat, long)$. In our problem we consider the following input matrices:

Definition 1 (Check-in matrix X). *"Based on the users' data we construct a binary check-in matrix $X \in \{0,1\}^{n \times m}$."*

Definition 2 (Social link matrix A). *"According to each user u's social relationships in $\mathcal{A}_u$, we compute a binary adjacency matrix $A \in \{0,1\}^{n \times n}$."*

Definition 3 (Geographical similarity matrix G). *"Given the geographical coordinates $(lat, long)$, we first compute the angular distance $\delta(a,b)$ between each pair of POIs a and b based on the Haversine formula[1], and then we calculate the geographical similarity matrix $G \in \mathbb{R}_+^{m \times m}$. Each element of G is computed as $G(a,b) = \frac{1}{1+(\delta(a,b) \times r)}$, with $r = 6,371\,\text{km being the earth radius.}$"*

Definition 4 (Sequential transition matrix T). *"Provided that users' check-in data are timestamped, we calculate a transition matrix $T \in \mathbb{R}_+^{m \times m}$, where each element $T(a,b)$ corresponds to the frequency of successive POI visits, $a \rightarrow b$. In the sequential transition matrix T, we filter out successive POI visits in a long interval e.g., more than a day, as these successive POI visits are weakly or not correlated at all [26]."*

Given user preferences in the check-in matrix X, users' contextual information in A and POIs' contextual information in G and T, the goal of our model is to generate top-k POI recommendations for a user $u \in \mathcal{N}$. In our GeoDCF model we formulate the POI recommendation problem as a pairwise ranking task [17]. We define a check-in probability[2] x_{ui}, where $x_{ui} = X(u,i)$ denotes that user u has already visited POI i. Thus, we can define two disjoint sets, a set $\mathcal{X}_u^+$ of visited POIs that user u has already checked-in, and a set $\mathcal{X}_u^-$ of non-visited POIs. For the task of POI recommendation, we build a pairwise ranking model that is able to rank the visited POIs before the non-visited ones. For any pair of POIs i and j, with $i \in \mathcal{X}_u^+$ and $j \in \mathcal{X}_u^-$, the check-in probability x_{ui} should be greater than x_{uj}. To describe this relation we define a *partial relation* $i >_u j$. For each user $u \in \mathcal{N}$ the set of all partial relationships is computed as follows:

$$\mathcal{R}_u = \{i >_u j | i \in \mathcal{X}_u^+, j \in \mathcal{X}_u^-\} \quad (1)$$

We define our POI recommendation task as the following ranking problem:

Definition 5 (Problem). *"Given the set of all partial relationships $\mathcal{R}_u$ for each user $u \in \mathcal{N}$, the goal of GeoDCF is to maximize the ranking likelihood probability as follows:"*

$$\max \prod_{u \in \mathcal{N}} \prod_{(i,j) \in \mathcal{R}_u} P(i >_u j) \quad (2)$$

4 The GeoDCF Model

4.1 Model Overview

An overview of the proposed GeoDCF model is presented in Fig. 1. The inputs are the check-in matrix X and contextual matrices A, G and T (Sect. 3). In the

[1] https://en.wikipedia.org/wiki/Haversine_formula.

[2] Initially, the check-in probability is binary to construct matrix X, and after the model learning the entries of X are in the range of $[0,1]$.

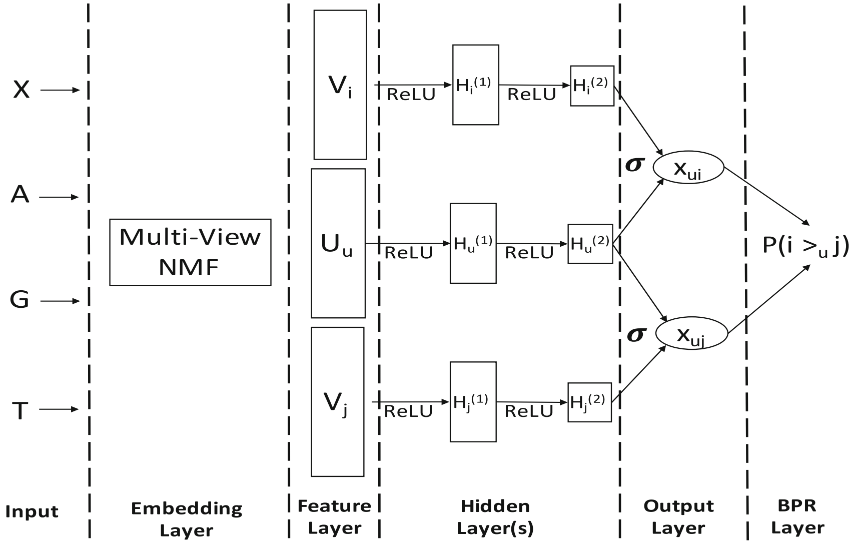

Fig. 1. Overview of GeoDCF. In this example, we use $h = 2$ hidden layers in our deep collaborative filtering strategy. For each user u POIs i and j denote a visited and a non-visited POI, respectively.

embedding layer the goal is to jointly learn the influence of the contextual information on user preferences and compute the latent matrices $U \in \mathbb{R}^{n \times d}$ and $V \in \mathbb{R}^{m \times d}$ of the preference matrix X, with d being the low dimensional embeddings. In the remaining layers of our architecture in Fig. 1, we perform *BPR learning* for the pairwise ranking task to generate POI recommendations. As defined in our pairwise ranking task in Eq. (2), for each user u we have pairs of partial relations $(i, j) \in \mathcal{R}_u$. In the **feature layer** we consider the POI latent vectors $V_i \in \mathbb{R}^d$ and $V_j \in \mathbb{R}^d$, that is the i-th and j-th rows of V, as well as the user latent vector $U_u \in \mathbb{R}^d$, the u-th row of U. Then, we design three neural networks[3], where each latent vector V_i, U_u and V_j is provided to the respective neural network. Given h **hidden layers**, we first try to capture the non-linear representations $H_i^{(q)}$, $H_u^{(q)}$ and $H_j^{(q)}$ of V_i, U_u and V_j in each neural network separately, with $q = 1, \ldots, h$. In the example of Fig. 1 we use $h = 2$ hidden layers. The **output layer** calculates the check-in probabilities x_{ui} and x_{uj} by combining the last hidden layers $H_i^{(h)}$, $H_u^{(h)}$ and $H_j^{(h)}$ with a sigmoid function $\sigma(x) = 1/(1 + e^{-x})$. Finally, the **BPR layer** predicts the probability of the partial relation $P(i >_u j)$.

[3] Alternatively, we could concatenate the three latent vectors and use a single neural network, increasing the computational cost of the learning process.

4.2 Embedding Layer

Given that we have to learn the influence of the contextual matrices A, G and T on user preferences in the check-in matrix X, at the embedding layer we formulate a multi-view joint factorization problem. In particular, we define the following joint loss function:

$$\min_{\mathbf{\Theta}_e} \mathfrak{L} = \mathfrak{L}_X + \lambda_A \mathfrak{L}_A + \lambda_G \mathfrak{L}_G + \lambda_T \mathfrak{L}_T \tag{3}$$

where the four loss functions $\mathfrak{L}_X$, $\mathfrak{L}_A$, $\mathfrak{L}_G$ and $\mathfrak{L}_T$ correspond to the joint factorizations of the input matrices X, A, G and T. $\mathbf{\Theta}_e$ is the parameter set of the joint loss function $\mathfrak{L}$, and parameters λ_A, λ_G and λ_T regularize the respective loss functions in Eq. (3). Note that in Eq. (3) a regularization parameter for $\mathfrak{L}_X$ is omitted, as matrix X is the main check-in matrix with user preferences. The problem of the joint loss function in Eq. (3) is similar with the Multi-View Non-negative Matrix Factorization (MV-NMF) problem of [5]. MV-NMF tries to bring the latent matrices of different views as close as possible to a common consensus matrix. For example, if we assume that the four input matrices are only coupled at the POI dimension, we have a consensus matrix $V^* \in \mathbb{R}^{m \times d}$, with d being the low-dimensional latent embeddings. While jointly factorizing the input matrices, the goal of MV-NMF is to minimize the four reconstruction errors $||V^{(v)} - V^*||_F^2$ of the consensus matrix V^* and the respective POI latent matrices $V^{(v)} \in \mathbb{R}^{m \times d}$, with $v = 1, \ldots, 4$. Instead of having couplings at one dimension as in [5], in our setting the input matrices might be coupled at different dimensions, that is either at the user or POI dimensions. Thus, we extend [5], by introducing the user and POI consensus matrices $U^* \in \mathbb{R}^{n \times d}$ and $V^* \in \mathbb{R}^{m \times d}$ for the couplings at the user and POI dimensions, accordingly. We calculate the loss functions $\mathfrak{L}_X$, $\mathfrak{L}_A$, $\mathfrak{L}_G$ and $\mathfrak{L}_T$ of Eq. (3) as follows:

- $\mathfrak{L}_X = ||X - UV^\top||_F^2 + \gamma_X ||U - U^*||_F^2 + \delta_X ||V - V^*||_F^2$, with the check-in matrix X being coupled with all the contextual matrices at the user or POI dimensions. $U \in \mathbb{R}^{n \times d}$ and $V \in \mathbb{R}^{m \times d}$ are the user and item latent matrices, when factorizing X.
- $\mathfrak{L}_A = ||A - U_A V_A^\top||_F^2 + \gamma_A ||U_A - U^*||_F^2$. The social link matrix A is only coupled with the check-in matrix X at the user dimension, thus the reconstruction error of the POI consensus matrix V^* is omitted. Provided that A is symmetric we preserve the latent matrix $U_A \in \mathbb{R}^{n \times d}$, with $U_A = V_A$.
- $\mathfrak{L}_G = ||G - U_G V_G^\top||_F^2 + \delta_G ||V_G - V^*||_F^2$. The geographical similarity matrix G is coupled with the check-in matrix X at the POI dimension. Given that G is symmetric we keep only the latent matrix $V_G \in \mathbb{R}^{n \times d}$, with $U_G = V_G$.
- $\mathfrak{L}_T = ||T - U_T V_T^\top||_F^2 + \delta_T ||V_T - V^*||_F^2$. The sequential transition matrix T is coupled with the check-in matrix X at the POI dimension. In this case, we also preserve the latent matrix $V_T \in \mathbb{R}^{m \times d}$, with $U_T = V_T$.

The regularization parameters γ_X and γ_A control the reconstruction errors of the respective user latent matrices of each loss function and the user consensus matrix U^*. Accordingly, parameters δ_X, δ_G and δ_T are used to regularize the reconstruction errors of the respective POI latent matrices of each loss function and the POI consensus matrix V^*. To reduce the complexity of our model, in our implementation we set the regularization parameters for the consensus matrices to 0.01.

Summarizing, the parameter set $\boldsymbol{\Theta}_e$ of the joint loss function $\mathfrak{L}$ in Eq. (3) is set to $\boldsymbol{\Theta}_e = \{U, V, U_A, V_G, V_T, U^*, V^*\}$, as A, G and T are symmetric matrices, with $U_A = V_A$, $U_G = V_G$ and $U_T = V_T$. However, the minimization problem of Eq. (3) is not convex with respect to all the variables of the parameter set $\boldsymbol{\Theta}_e$. To solve this problem, we follow an alternating optimization strategy, that is update one variable while fixing the remaining variables of $\boldsymbol{\Theta}_e$. According to the learning strategy of multiplicative rules [5], we compute the update rules of each variable for the alternating optimization algorithm. Due to lack of space we omit the presentation of the update rules, as they can be computed in a similar way as in the study at [5]. By solving the minimization problem of Eq. (3), the embedding layer computes the user and POI latent matrices U and V of the check-in matrix X with user preferences, by also accounting for the contextual information.

4.3 BPR Learning

Feature Layer. At the remaining layers of our architecture in Fig. 1 we adopt the BPR technique to produce top-k recommendations. Having computed the user and POI latent matrices U and V at the embedding layer, for each user $u \in \mathcal{N}$ we consider the partial relations $(i, j) \in \mathcal{R}_u$ based on Eq. (1). Then, in the feature layer we consider the low d-dimensional embeddings, that is the latent vectors V_i, U_u and V_j, which are then provided to the respective three neural networks, as shown in Fig. 1.

Hidden Layers. When training the GeoDCF model we aim to maximize the likelihood in Eq. (2), hence the loss function of GeoDCF becomes:

$$\min_{\boldsymbol{\Theta}_b} \mathfrak{L} = -\sum_{u \in \mathcal{N}} \sum_{(i,j) \in \mathcal{R}_u} P(i >_u j) + \lambda ||\boldsymbol{\Theta}_b||^2 \tag{4}$$

$\boldsymbol{\Theta}_b$ is the parameter set, with $\boldsymbol{\Theta}_b = \{W_i^{(q)}, W_u^{(q)}, W_j^{(q)}, b_i^{(q)}, b_u^{(q)}, b_j^{(q)}\}$, $\forall q = 1, \ldots, h$, where h is the number of hidden layers used in the three neural networks of Fig. 1. Matrices $W_i^{(q)}$, $W_u^{(q)}$ and $W_j^{(q)}$ are the weighting matrices of the q-th hidden layers to produce the deep learning representations of the latent vectors V_i, U_u and V_j. Variables $b_i^{(q)} b_u^{(q)}, b_j^{(q)}$ denote the respective biases of the q-th hidden layers of each neural network. As the size of hidden layers is important, in our architecture the bottom layer is the widest and each successive layer has a smaller number of hidden units. This way it learns more abstractive features of the d-dimensional embeddings and consequently better captures the non-linear correlations of the multifaceted contextual information with user preferences. For each neural network we implement the tower structure, halving the layer size for each successive layer. Hence, to implement the tower architecture we add the constraint of $2^h \leq d$ for the number of hidden layers h and the low d-dimensional embeddings of MV-NMF. For the hidden layers there are several choices of activation functions, like sigmoid, hyperbolic tangent $tanh(x)$ and rectifier linear unit function $ReLU(x)$. In our implementation, we used ReLU activation functions, with $ReLU(x) = \max(0, x)$, as they are non-saturated[4], well-suited for sparse data and making the model less likely to be overfitting [19]. Using ReLU activation functions, $\forall q = 1, \ldots, h$, the q-th hidden layers of the three neural networks produce the respective representations:

$$\begin{aligned} H_i^{(q)} &= ReLU(W_i^{(q)} H_i^{(q-1)} + b_i^{(q-1)}) \\ H_u^{(q)} &= ReLU(W_u^{(q)} H_u^{(q-1)} + b_u^{(q-1)}) \\ H_j^{(q)} &= ReLU(W_j^{(q)} H_j^{(q-1)} + b_j^{(q-1)}) \end{aligned} \tag{5}$$

with $H_i^{(0)} = V_i$, $H_u^{(0)} = U_u$ and $H_j^{(0)} = V_j$.

Output and BPR Layers. At the output layer, we use the hidden representations and the biases of the last hidden layers, that is the h-th layers of the three neural networks, which are then combined to compute the check-in probabilities x_{ui} and x_{uj} (Sect. 3). At the output layer we use the sigmoid function σ to ensure that the check-in probabilities x_{ui} and x_{uj} are in the range of $[0, 1]$. The check-in probabilities x_{ui} and x_{uj} are calculated as follows:

$$\begin{aligned} x_{ui} &= \sigma({H_i^{(h)}}^\top H_u^{(h)} + b_i^{(h)} + b_u^{(h)}) \\ x_{uj} &= \sigma({H_j^{(h)}}^\top H_u^{(h)} + b_j^{(h)} + b_u^{(h)}) \end{aligned} \tag{6}$$

Provided that x_{ui} and $x_{uj} \in [0, 1]$, at the BPR layer the partial relation between x_{ui} and x_{uj} is computed as $P(i >_u j) = (x_{ui} - x_{uj})/2 + 0.5$. Then, based on the computed probability $P(i >_u j)$, the prediction of a non-visited POI i is calculated by forwarding its low d-dimensional embedding V_i on the respective neural network as shown in Fig. 1 and then computing the check-in probability x_{ui}. The final top-k POI recommendations are generated by ranking the non-visited POIs based on the probability $P(i >_u j)$.

Model Training. In our implementation we used Tensorflow[5]. We computed the model parameters $\mathbf{\Theta}_b$ via backpropagation with stochastic gradient descent. In particular, we employed mini-batch Adam, which adapts the learning rate for each parameter by performing smaller updates for frequent and larger updates for infrequent parameters. In each backpropagation iteration we performed negative sampling, as defined in BPR, to randomly select a subset of non-visited POIs as negative instances $j \in \mathcal{X}_u^-$. In our implementation we sampled five negative samples for each positive/observed sample, and set the batch size of mini-batch Adam to 512 with a learning rate 1e−4. Finally, to account for the fact that

[4] The saturation problems occurs when neurons stop learning and their output is near either 0 or 1, a problem that can be suffered by the sigmoid and tanh functions [19].

[5] www.tensorflow.org.

the initialization of the model parameters $\boldsymbol{\Theta}_b$ plays an important role for the convergence and performance of our model, we followed a pretraining strategy. By applying single-view factorization of X and producing the respective latent matrices U and V, we first trained our model only using check-in data in X with random initializations until convergence - ignoring the contextual information in matrices A, G and T. Then, we used the trained parameters as the initialization of our model with the contextual information.

5 Experimental Evaluation

5.1 Datasets

In our experiments we used two publicly available datasets from Gowalla and Foursquare. The Gowalla check-in dataset[6] was generated from February 2009 to October 2010. Following [10,20] we filter out those users with fewer than 15 check-in POIs and those POIs with fewer than 10 visitors. The filtered dataset comprises 18,737 users, 32,510 POIs, 1,278,274 check-ins. The Gowalla check-in dataset includes all the contextual information, that is social correlation, as well as geographical and sequential transition information. The Foursquare dataset[7] includes check-in data from April 2012 to September 2013. We used the records generated within United States and eliminated those users with fewer than 10 check-in POIs, as well as those POIs with fewer than 10 visitors. The filtered dataset contains 24,941 users, 28,593 POIs and 1,196,248 check-ins. In the Foursquare dataset, geographical and sequential transition information is available, whereas users' social relations are missing.

5.2 Evaluation Protocol

To evaluate the top-k recommendation performance of the examined models we used the ranking-based metrics recall ($R@k$) and Normalized Discounted Cumulative Gain ($NDCG@k$). Recall $R@k$ is defined as the ratio of the relevant (checked-in) POIs in the top-k ranked list over all the relevant POIs for each user. The Normalized Discounted Cumulative Gain $NDCG@k$ metric considers the ranking of the relevant POIs in the top-k list. For each user the Discounted Cumulative Gain is defined as:

$$DCG@k = \sum_{l=1}^{k} \frac{2^{rel_l} - 1}{\log_2 l + 1} \tag{7}$$

where rel_l represents the relevance score of POI l, that is binary relevance in our case. We consider a POI as relevant if a user has checked-in, and irrelevant otherwise. $NDCG@k$ is the ratio of $DCG@k$ over the ideal $iDCG@k$ value for each user, that is the $DCG@k$ value given the check-in data in the test set.

[6] http://snap.stanford.edu/data/loc-gowalla.html.

[7] https://sites.google.com/site/yangdingqi/home/foursquare-dataset.

Following the evaluation protocol of [9,20] we randomly select a percentage of 20% of the check-in data as a test set, while the remaining check-in data are used to train our model. We repeated our experiments five times, and in our results we report average recall and $NDCG$ over the five runs.

5.3 Compared Methods

In our experiments we compare the following methods:

- ***RankGeoFM*** [9]: a ranking-based model that first learns users' preference rankings for POIs, and then includes the geographical influence of neighboring POIs to generate top-k POI recommendations.
- ***USG*** [21]: a POI recommendation algorithm that considers both geographical influence and users' social correlation, following a friend-based collaborative filtering strategy with a pointwise loss function.
- ***PACE*** [20]: a deep learning strategy for jointly learning the embeddings of users and POIs to predict user preferences over POIs and all the available contextual information with a pointwise loss function.
- ***MV-NMF:*** a variant of our model, which ignores the deep learning strategy of GeoDCF by only performing multi-view NMF of the check-in matrix with the contextual information, as presented in Sect. 4.2. To generate recommendations we compute the factorized matrix as the product of the user and latent matrices $UV^\top$, and sort each row/user of the factorized matrix in a descending order. MV-NMF exploits all the available contextual information, and is a pointwise method.
- ***GeoDCF:*** the proposed model that first performs MV-NMF to calculate the user and POIs latent vectors and then performs BPR learning with our deep learning strategy.

The parameters of the examined methods have been determined via cross-validation and in our experiments we report the best results. The parameter analysis of the proposed method is further studied in Sect. 5.6.

5.4 Comparison with State-of-the-Art

In Fig. 2 we evaluate the performance of the examined models in terms of recall $R@k$ and $NDCG@k$, when varying the top-k POI recommendations. RankGeoFM and USG perform differently in the Gowalla and Foursquare datasets. Although both RankGeoFM and USG exploit users' check-in data and geographical information, in the Gowalla dataset USG achieves a better recommendation accuracy than RankGeoFM as USG also uses the available contextual information of users' social relations. Instead, users' social relations are missing from the Foursquare dataset (Sect. 5.1). As we can observe from Fig. 2 in the Foursquare dataset RankGeoFM beats USG. This occurs because RankGeoFM is a ranking-based method focusing on the ranking performance, while USG is a pointwise method. Regarding the most competitive method of PACE,

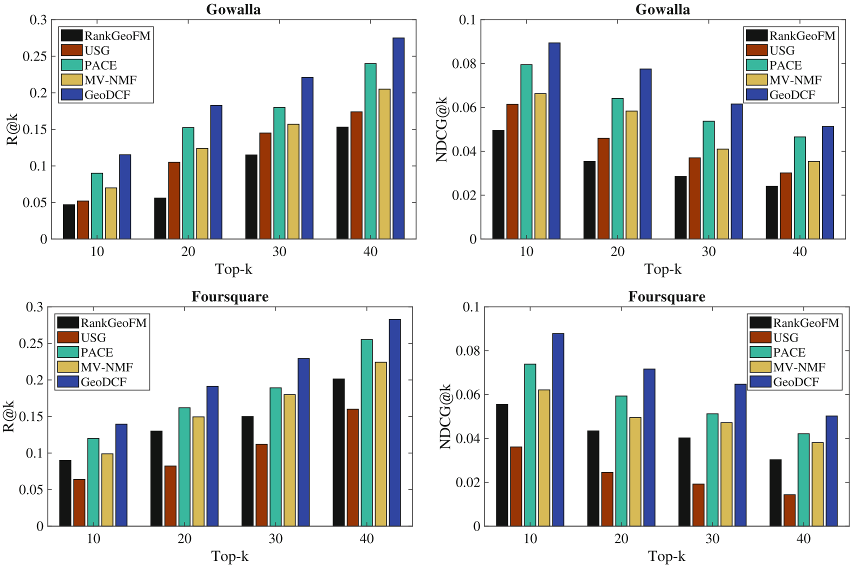

Fig. 2. Performance evaluation in terms of recall ($R@k$) and Normalized Discounted Cumulative Gain ($NDCG@k$) for the Gowalla and Foursquare datasets. Using the paired t-test, the proposed GeoDCF model outperforms all the baselines for $p < 0.05$.

Fig. 2 shows that PACE outperforms both RankGeoFM and USG by capturing the non-linear correlations of the available contextual information with its deep learning strategy. Compared to the proposed GeoDCF model, our MV-NMF variant performs poorly, as MV-NMF neither captures well the non-linear correlations of the users' and POIs' contextual information nor focuses on the ranking performance. Using the paired t-test we found out that compared to the second best method of PACE, our GeoDCF model achieves an average improvement of 18.96% and 17.81% in terms of recall and NDCG in all runs, at a significance level of $p < 0.05$. This occurs because PACE is a pointwise method and GeoDCF is a ranking-based model aiming to improve the top-k recommendation accuracy. Furthermore, GeoDCF also captures the non-linear correlations of the multifaceted contextual information with user preferences in our deep learning architecture.

5.5 Influence of Users' and POIs' Context

In Fig. 3 we evaluate separately the influence of users' and POIs' contextual information on our GeoDCF model. We denote *"check-in data"*, when GeoDCF only uses the check-in data to produce recommendations, ignoring any contextual information. Accordingly, *"check-in data+user context"* is a variant of the GeoDCF model which exploits check-in data and user context, that is users' social relations. Model *"check-in data+POI context"* is our variant when check-in

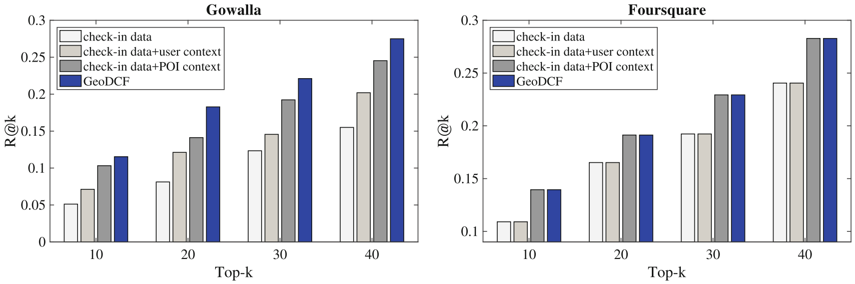

Fig. 3. Influence of users' and POIs' contextual information. Provided that in the Foursquare dataset users' social relations are missing, the variants *"check-in data"* and *"check-in data+user context"* have equal performance, and the variant *"check-in data+POI context"* has the same performance with GeoDCF.

data are only combined with POIs' contextual information, that is geographical and sequential transition information. As in the Foursquare dataset users' social correlations are missing, the variants *"check-in data"* and *"check-in data+user context"* have equal performance, and the variant *"check-in data+POI context"* has the same performance with GeoDCF. Clearly, as we can observe from Fig. 3 in both datasets the *"check-in data"* variant has the lowest performance, as it does not combine any contextual information with user preferences. This means that the contextual information of users or POIs can boost the recommendation accuracy. An interesting observation is that in the Gowalla dataset the *"check-in data+POI context"* variant outperforms the *"check-in data+user context"* variant, which indicates that POIs' context is more important than users' context in the POI recommendation task. This observation also complies with the observations of relevant studies such as [7,20].

5.6 Parameter Analysis

The two most important parameters in our GeoDCF model are: (i) the number of low dimensional embeddings d at the embedding layer; and (ii) the number of hidden layers h of the neural networks. Given the constraint of $2^h \leq d$ of Sect. 4.3, we vary the number of low dimensional embeddings d to the power of 2. For $d = [1024, 512, 256]$ we vary the number of hidden layers h from 1 to 5 by a step of 1. As described in Sect. 4.3 the bottom layer with the low dimensional embeddings d is the widest and each successive layer has a smaller number of hidden units, to learn more abstractive features of the d-dimensional embeddings. To better capture the non-linear correlations of the multifaceted contextual information, we implement the tower structure for each neural network, that is halving the layer size for each successive layer. For example, for $d = 1024$ and $h = 3$ we have the following tower architecture $1024 \rightarrow 512 \rightarrow 256 \rightarrow 128$ or for $d = 512$ and $h = 2$ we have the architecture of $512 \rightarrow 256 \rightarrow 128$. Figure 4 shows the impact of the different deep learning architectures. We observe that the best architecture is when $d = 256$ and $h = 3$ in the Gowalla dataset, and $d = 512$ and $h = 4$ in the Foursquare dataset, corresponding to the following architectures: $256 \rightarrow 128 \rightarrow 64 \rightarrow 32$ and $512 \rightarrow 256 \rightarrow 128 \rightarrow 64 \rightarrow 32$, respectively. For different d and h values GeoDCF cannot capture well the non-linear correlations of the multifaceted contextual information with users' check-in data, which explains the low performance of GeoDCF in these cases.

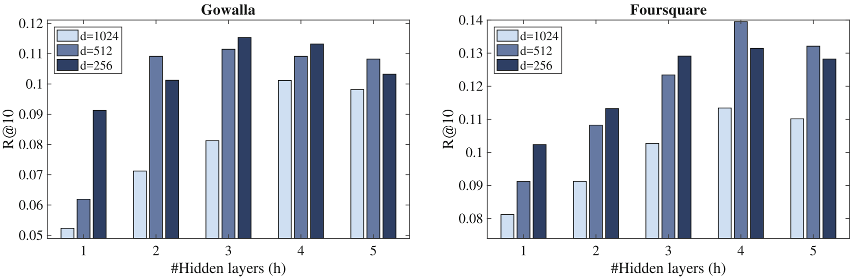

Fig. 4. Impact of different deep learning architectures when varying the number of low dimensional embeddings d at the embedding layer and the number of hidden layers h of the neural networks, subject to the constraint of $2^h \leq d$.

6 Conclusions

In this paper we presented GeoDCF, an efficient POI recommendation strategy to exploit the multifaceted information of users and POIs. The three key factors of the proposed model are the (i) exploit of the contextual information of users and POIs with a multi-view strategy at the embedding layer; (ii) capture of the non-linear correlations of the multifaceted contextual information with users' check-in data in our deep learning architecture; (iii) adding of a BPR layer at the top of our architecture to focus on the ranking performance. Our experimental evaluation on two benchmark datasets showed the superiority of GeoDCF to recently proposed baselines. Compared to the second best method, the proposed GeoDCF model achieved an average improvement of 18.96% and 17.81% in terms of recall and NDCG in all runs. We also evaluated GeoDCF with a variant of our model, which ignores the proposed deep learning architecture. Our experimental results demonstrated that GeoDCF outpeformed its variant. Clearly, the deep learning strategy can significantly boost the recommendation accuracy, by capturing the non-linear correlations of the contextual information and focusing on the ranking performance in the POI recommendation task. Finally, we evaluated the impact of users' and POIs' contextual information on our model separately. We showed that POIs' context contains more valuable information than users' social relations when generating POI recommendations, also confirmed by relevant studies [7,20]. Nowadays, users open multiple accounts on different social media platforms. An interesting future direction is to exploit user data from various social media platforms, following cross-domain strategies to produce POI recommendations. This is a challenging task as users behave differently in distinct social media platforms. For example, we plan to extend our GeoDCF model to generate POI recommendations for Foursquare users based on user data from Twitter and Instagram [15,16].

References

1. Cheng, C., Yang, H., Lyu, M.R., King, I.: Where you like to go next: successive point-of-interest recommendation. In: Proceedings of the International Joint Conference on Artificial Intelligence, pp. 2605–2611 (2013)
2. Cho, E., Myers, S.A., Leskovec, J.: Friendship and mobility: user movement in location-based social networks. In: Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining, pp. 1082–1090 (2011)
3. Ding, D., Zhang, M., Li, S., Tang, J., Chen, X., Zhou, Z.: BayDNN: friend recommendation with Bayesian personalized ranking deep neural network. In: Proceedings of the ACM International Conference on Information and Knowledge Management, pp. 1479–1488 (2017)
4. Farseev, A., Samborskii, I., Filchenkov, A., Chua, T.: Cross-domain recommendation via clustering on multi-layer graphs. In: Proceedings of the ACM International Conference on Research and Development in Information Retrieval, pp. 195–204 (2017)
5. Gao, J., Han, J., Liu, J., Wang, C.: Multi-view clustering via joint nonnegative matrix factorization. In: Proceedings of the SIAM International Conference on Data Mining, pp. 252–260 (2013)
6. Hu, Y., Koren, Y., Volinsky, C.: Collaborative filtering for implicit feedback datasets. In: Proceedings of the IEEE International Conference on Data Mining, pp. 263–272 (2008)
7. Li, H., Ge, Y., Hong, R., Zhu, H.: Point-of-interest recommendations: learning potential check-ins from friends. In: Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining, pp. 975–984 (2016)
8. Li, S., Kawale, J., Fu, Y.: Deep collaborative filtering via marginalized denoising auto-encoder. In: Proceedings of the ACM International Conference on Information and Knowledge Management, pp. 811–820 (2015)
9. Li, X., Cong, G., Li, X., Pham, T.N., Krishnaswamy, S.: Rank-GeoFM: a ranking based geographical factorization method for point of interest recommendation. In: Proceedings of the ACM International Conference on Research and Development in Information Retrieval, pp. 433–442 (2015)
10. Lian, D., Zhao, C., Xie, X., Sun, G., Chen, E., Rui, Y.: GeoMF: joint geographical modeling and matrix factorization for point-of-interest recommendation. In: Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining, pp. 831–840 (2014)
11. Liu, Y., Wei, W., Sun, A., Miao, C.: Exploiting geographical neighborhood characteristics for location recommendation. In: Proceedings of the ACM International Conference on Information and Knowledge Management, pp. 739–748 (2014)
12. Manotumruksa, J., Macdonald, C., Ounis, I.: A personalised ranking framework with multiple sampling criteria for venue recommendation. In: Proceedings of the ACM International Conference on Information and Knowledge Management, pp. 1469–1478 (2017)

13. Rafailidis, D., Crestani, F.: Collaborative ranking with social relationships for top-n recommendations. In: Proceedings of the ACM International Conference on Research and Development in Information Retrieval, pp. 785–788 (2016)
14. Rafailidis, D., Crestani, F.: Joint collaborative ranking with social relationships in top-n recommendation. In: Proceedings of the ACM International Conference on Information and Knowledge Management, pp. 1393–1402 (2016)
15. Rafailidis, D., Crestani, F.: Top-n recommendation via joint cross-domain user clustering and similarity learning. In: Frasconi, P., Landwehr, N., Manco, G., Vreeken, J. (eds.) ECML PKDD 2016. LNCS (LNAI), vol. 9852, pp. 426–441. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46227-1_27
16. Rafailidis, D., Crestani, F.: A collaborative ranking model for cross-domain recommendations. In: Proceedings of the ACM International Conference on Information and Knowledge Management, pp. 2263–2266 (2017)
17. Rendle, S., Freudenthaler, C., Gantner, Z., Schmidt-Thieme, L.: BPR: Bayesian personalized ranking from implicit feedback. In: Proceedings of the International Conference on Uncertainty in Artificial Intelligence, pp. 452–461 (2009)
18. Wang, H., Wang, N., Yeung, D.: Collaborative deep learning for recommender systems. In: Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining, pp. 1235–1244 (2015)
19. Xu, L., Choy, C., Li, Y.: Deep sparse rectifier neural networks for speech denoising. In: Proceedings of the IEEE International Workshop on Acoustic Signal Enhancement, pp. 1–5 (2016)
20. Yang, C., Bai, L., Zhang, C., Yuan, Q., Han, J.: Bridging collaborative filtering and semi-supervised learning: a neural approach for POI recommendation. In: Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining, pp. 1245–1254 (2017)
21. Ye, M., Yin, P., Lee, W., Lee, D.L.: Exploiting geographical influence for collaborative point-of-interest recommendation. In: Proceeding of the ACM International Conference on Research and Development in Information Retrieval, pp. 325–334 (2011)
22. Ying, H., Chen, L., Xiong, Y., Wu, J.: Collaborative deep ranking: a hybrid pairwise recommendation algorithm with implicit feedback. In: Bailey, J., Khan, L., Washio, T., Dobbie, G., Huang, J.Z., Wang, R. (eds.) PAKDD 2016. LNCS (LNAI), vol. 9652, pp. 555–567. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-31750-2_44
23. Yuan, F., Jose, J.M., Guo, G., Chen, L., Yu, H., Alkhawaldeh, R.S.: Joint geospatial preference and pairwise ranking for point-of-interest recommendation. In: Proceedings of the IEEE International Conference on Tools with Artificial Intelligence, pp. 46–53 (2016)
24. Zhang, J., Chow, C.: iGSLR: personalized geo-social location recommendation: a kernel density estimation approach. In: Proceeding of the ACM International Conference on Advances in Geographic Information Systems, pp. 324–333 (2013)
25. Zhang, J., Chow, C.: GeoSoCa: exploiting geographical, social and categorical correlations for point-of-interest recommendations. In: Proceedings of the ACM International Conference on Research and Development in Information Retrieval, pp. 443–452 (2015)
26. Zhang, J., Chow, C., Li, Y.: LORE: exploiting sequential influence for location recommendations. In: Proceedings of the ACM International Conference on Advances in Geographic Information Systems, pp. 103–112 (2014)