Neural Personalized Ranking for Image Recommendation

Wei Niu, James Caverlee, Haokai Lu
Department of Computer Science and Engineering, Texas A&M University
{wei,caverlee,hlu}@cse.tamu.edu

ABSTRACT

We propose a new model toward improving the quality of image recommendations in social sharing communities like Pinterest, Flickr, and Instagram. Concretely, we propose *Neural Personalized Ranking (NPR)* – a personalized pairwise ranking model over implicit feedback datasets – that is inspired by Bayesian Personalized Ranking (BPR) and recent advances in neural networks. We further build an enhanced model by augmenting the basic NPR model with multiple contextual preference clues including user tags, geographic features, and visual factors. In our experiments over the Flickr YFCC100M dataset, we demonstrate the proposed NPR model is more effective than multiple baselines. Moreover, the contextual enhanced NPR model significantly outperforms the base model by 16.6% and a contextual enhanced BPR model by 4.5% in precision and recall.

ACM Reference Format:
Wei Niu, James Caverlee, Haokai Lu. 2018. Neural Personalized Ranking for Image Recommendation. In *Proceedings of 11th ACM International Conf. on Web Search and Data Mining (WSDM 2018).* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3159652.3159728

1 INTRODUCTION

One of the foundations of many web and app-based communities is *image sharing*. For example, Pinterest, Facebook, Twitter, Flickr, Instagram, and Snapchat all enable communities to share, favorite, re-post, and curate images. And yet, these social actions are far outnumbered by the total number of images in the system; that is, there may be many valuable images undiscovered by each user. Hence, considerable research has focused on the challenge of *image recommendation* in these communities, e.g,. [8, 15, 19, 20, 23, 24, 31].

However, many of these works mainly leverage user profile and behavior patterns. Due to the extreme sparsity of user feedback in image sharing communities and a lack of proper representation, traditional recommendation including collaborative filtering and content-based methods face challenges. In contrast, Bayesian Personalized Ranking (BPR) has shown state-of-the-art performance for recommendation in implicit feedback datasets [30]. Yet, there exists some limitations: (i) First, user preferences in BPR are calculated as the inner product of user latent vectors and image latent vectors, which assigns equal weight to each dimension of the latent feature space, meaning the variability of user preferences may not be adequately captured; (ii) Second, the matrix factorization component of BPR is linear in nature, which has limited expressiveness when compared to nonlinear methods; and (iii) existing efforts for distributed BPR typically use partially shared memory which may limit its scalability.

To overcome these challenges, we propose *Neural Personalized Ranking (NPR)* – a new neural network based personalized pairwise ranking model for implicit feedback, which incorporates the idea of generalized matrix factorization. Neural models promise potentially more flexibility in model design, added nonlinearity through activations, and ease of parallelization. While recent work in neural methods for recommendation has focused on modeling side information [36, 40] or building pointwise learning models by directly modeling user ratings [10], a key feature of NPR is its careful modeling of users' implicit feedback via a relaxed assumption about unobserved items using pairwise ranking that builds on top of neural network based generalized matrix factorization components. Further, to alleviate the sparsity of user feedback and improve the quality of recommendation, we propose to leverage multiple categories of contextual information. Correspondingly, we augment the baseline NPR model with multiple contextual preference clues for deriving *Contextual Neural Personalized Ranking (C-NPR)* to better uncover user preferences. In particular, these preference clues include user tags, geographic features, and visual factors.

In our experiments over the Flickr YFCC100M dataset, we demonstrate the proposed NPR model's effectiveness in comparison to several state-of-the-art approaches. Moreover, the contextual enhanced NPR model significantly outperforms the baseline model by 16.6% and a contextual-BPR model by 4.5% in precision and recall. We find that NPR is more effective than BPR when there is inadequate training data.

2 RELATED WORK

Research attention on recommendation has shifted towards the common scenario where only implicit feedback is available, as is common in social imaging sharing communities. One pioneer work terms such a scenario as one-class collaborative filtering [29], where the authors proposed to weight positive and unobserved feedback differently in fitting the objective function. This idea was further improved to introduce varying confidence levels [13]. These approaches are mainly variations of pointwise approaches suitable for explicit feedback. Pairwise learning for implicit feedback, specifically Bayesian personalized ranking with matrix factorization (BPR-MF), typically outperforms pointwise learning counterparts [30].

Image Recommendation. Many works have tackled the problem of image recommendation, e.g., [8, 15, 19]. For example, Jing et al. use a weighted matrix factorization model that combines image importance and local community user rating [15]. Sang et al.

WSDM 2018, February 5–9, 2018, Marina Del Rey, CA, USA

ACM ISBN 978-1-4503-5581-0/18/02…$15.00
https://doi.org/10.1145/3159652.3159728

measure the distance of an image and a personalized query language through a graph-based topic-sensitive probabilistic model [31]. Later works begin incorporating a variety of visual features, including high-level features from deep convolutional neural networks. For example, Liu et al. introduce social embedding image distance learning that learns image similarity based on social constraints and leverages Borda Count for recommendation [23]. Lei et al. propose a comparative deep learning model that learns image and user representation jointly and identifies the nearest neighbor images of each user for recommendation [18].

Context-aware Recommendation. To overcome ratings sparsity, many recommenders have proposed to incorporate additional contextual information [1], including but not limited to social connections [14, 25], content [26, 35], and so on. Visual features have received much attention in recent work, with some methods using metrics for visual similarity according to social behavior or activity pattern to identify compatible items [23, 27] and visual enhanced recommendation [9]. With the rapid growth of location-based social networks and smart mobile devices, many applications take advantage of geographical information in modeling video watching preferences [3], Yelp ratings prediction [12], and most commonly in point of interest recommendation, where representative work includes [4, 21, 22, 39, 41]. In our work, we derive and integrate multiple categories of contextual features for image recommendation. We show that our proposed method to model user's preference is effective and adaptable to different frameworks.

Deep recommendation & ranking with implicit feedback. Recently, we have seen increasing efforts devoted to recommendation models based on deep learning [5–7, 10, 11, 18, 34, 37]; note that we neglect discussion of works that leverage deep learning for deriving features then can be integrated into traditional recommendation models. Several of these target the common scenario of implicit feedback [10, 18, 34]. For example, He and et al. introduce a pointwise neural collaborative filtering framework which includes an ensemble of multi-layer perceptron and generalized matrix factorization components that jointly contribute to better performance [10]. The work that is most relevant to ours is [34], where the authors propose a multi-layer feed forward neural network based pairwise ranking model which can be applied to personalized recommendation. Distinct from previous works, we propose a pairwise ranking based recommendation model that incorporates the idea of generalized matrix factorization for implicit feedback. We also provide a framework for explicitly modeling user's contextual preference for alleviating sparsity issues.

3 PRELIMINARIES

Our goal is to provide *personalized image recommendation*, such that each user is recommended a personalized list of images.

Problem Statement. Formally, we assume a set of M users $\mathcal{U}$=$\{u_1, u_2, ..., u_M\}$ and a set of N images $\mathcal{I}$=$\{i_1, i_2, ..., i_N\}$. We further assume some users have explicitly expressed their interest for a subset of the images in $\mathcal{I}$. This preference may be in the form of a "like" or similar social sharing function. We aim to recommend for each user a personalized list of images from the set $\mathcal{I}$.

3.1 Matrix Factorization

Toward tackling the problem of *personalized image recommendation*, we begin with a straightforward adaptation of latent factor matrix factorization (MF) [17]. The standard formulation is: the preference r_{ui} of a user u towards an image i is predicted as:

$$r_{ui} = \boldsymbol{p}_u^T \boldsymbol{q}_i + b_u + b_i + \alpha \tag{1}$$

where $\boldsymbol{p}_u$ and $\boldsymbol{q}_i$ are the K-dimensional latent factors of user preference and image characteristics, respectively. The inner product $\boldsymbol{p}_u^T \boldsymbol{q}_i$ of the user latent vector and image latent vector represents a user's preference towards an image; it measures how well the user preferences align with the properties of the image. b_u and b_i correspond to user and image bias terms while α is a global offset.

3.2 Bayesian Personalized Ranking

Since users only provide sparse one-class positive feedback (the "likes"), there is ambiguity in the interpretation of non-positive images since the negative examples and unlabeled positive examples are mixed together [29]. In this implicit feedback scenario, we may only assume users prefer the liked images to those that are not acted upon. To estimate the latent factors, instead of trying to model the matrix of "likes" directly in a pointwise setting with a least square regression formulation, we can construct the learning objective based on pairwise ranking between images. This idea is key to Bayesian Personalized Ranking [30], such that observed likes should be ranked higher than the unobserved ones. The model then tries to find latent factors that can be used to predict the expected preference of a user for an item.

Formally, we can adapt BPR to the *personalized image recommendation* task as follows. Suppose we have a user u_h and a pair of images i_j and i_k. User u_h's feedback for i_j is positive, and feedback for i_k is unobserved: we denote this relation as $j >_h k$. BPR aims to maximize the posterior probability $p(\Theta | j >_h k)$, where Θ is the set of parameters we try to estimate. According to Bayes' rule:

$$p(\Theta | j >_h k) \propto p(j >_h k | \Theta) P(\Theta)$$

and the likelihood function is defined as:

$$p(j >_h k | \Theta) = \delta(r_{hj} - r_{hk})$$

where $\delta(\cdot)$ is the sigmoid function. To simplify notation, We will use the index of a user and an image. We assume a Gaussian prior $\Theta \sim N(0, \lambda_\Theta I)$, where λ_Θ is a set of model-specific parameters and I is the identity matrix. The prior provides regularization for the parameters to prevent overfitting.

Our objective is to find Θ that maximizes the log-likelihood for all users and all images:

$$\arg\max_{\Theta} \sum_{u_h \in \mathcal{U}, i_j \in \mathcal{P}_h, i_k \in \mathcal{N}_h} \left(\ln \left(\delta(r_{hj} - r_{hk}) \right) - \lambda_\Theta \left\| \Theta_{hjk} \right\|^2 \right)$$

where $\mathcal{P}_h$, $\mathcal{N}_h$ are the sets of images for which u_h has provided positive feedback and u_h's feedback is unobserved, respectively. Θ is $\{p_u, q_i, b_i\}$ for all users and images. With this pairwise setting, the user bias and global offset in Equation 1 cancel out.

4 NEURAL PERSONALIZED RANKING

In this section, we seek to complement existing matrix factorization and BPR-based approaches to personalized image recommendation

through the exploration of a new neural network based personalized pairwise ranking model. Neural recommendation models promise some exciting characteristics in comparison with BPR: (i) First, user preferences in BPR are calculated as the inner product of user latent vector and image latent vector, which assigns equal weight to each dimension of the latent feature space. In contrast, neural methods may be able to capture the variability of user preferences by relaxing this equal weight requirement. (ii) Second, the matrix factorization component of BPR is linear in nature, which has limited expressiveness. In contrast, neural methods offer more flexibility by adding nonlinearity through activations. (iii) Finally, many neural methods may be easily parallelized for scalable computation, whereas existing work on distributed BPR typically uses partially shared memory which may limit its scalability. In summary, neural models promise potentially more flexibility in model design, added nonlinearity through activations, and ease of parallelization.

4.1 Model Architecture

The NPR model structure is shown in Figure 1. There are three inputs to the model, the user and a pair of images, represented as tuple of index (h, j, k). Then user and image indexes are one-hot encoded as tuple of vectors $(\boldsymbol{u}_h, \boldsymbol{i}_j, \boldsymbol{i}_k)$. Since there are M users and N images, the dimensions of $\boldsymbol{u}_h, \boldsymbol{i}_j, \boldsymbol{i}_k$ are M, N, and N respectively. The output of the proposed model is the ground truth value which we train the model against:

$$g(h,j,k) = \begin{cases} 1 & \text{for } j >_h k \\ -1 & \text{for } j <_h k \end{cases}$$

where $j >_h k$ denotes that user u_h prefers image i_j to i_k. This definition transforms the ranking problem into a binary classification problem, which aims to check whether the pairwise preference relation holds. Following the input layer, each input is fully connected to the corresponding embedding layer for the sake of learning a compact representation of the users and images. The embedding dimension for both users and images are the same. We denote the embeddings as $\boldsymbol{p}_h$, $\boldsymbol{q}_j$, $\boldsymbol{q}_k$. Formally,

$$\boldsymbol{p}_h = \boldsymbol{W}_u \boldsymbol{u}_h, \quad \boldsymbol{q}_j = \boldsymbol{W}_i \boldsymbol{i}_j, \quad \boldsymbol{q}_k = \boldsymbol{W}_i' \boldsymbol{i}_k.$$

where $\boldsymbol{W}_u$,$\boldsymbol{W}_i$, $\boldsymbol{W}_i'$ are embedding matrices for users and images. As the model architecture is vertically symmetric, let's focus on the substructure marked inside the dotted triangle (see Figure 1). In the merge layer, user and image embedding vectors are multiplied element-wise, such that each dimension of the user preference vector and corresponding image properties are in line. This step is analogous to traditional matrix factorization. The resulting vector has the same dimension as the embeddings. More precisely:

$$\boldsymbol{m}_{hj} = \boldsymbol{p}_h \circ \boldsymbol{q}_j$$

where $\circ$ denotes the element-wise product. The merge layer is then connected to a single neuron dense layer, which computes the weighted sum of all dimensions and passes it through a ReLU nonlinear activation. Compared to traditional matrix factorization, such a design allows each latent dimension to vary in importance and supports additional expressiveness through non-linearity. We adopt ReLU here based on our exploratory experiments, where we find that alternative activation functions like *sigmoid* and *tanh* suffer from saturation, which leads to overfitting. The output is preference score r_{hj}:

$$r_{hj} = a(\boldsymbol{w}^T \boldsymbol{m}_{hj} + b_1)$$

where $a(\cdot)$ is the activation function, $\boldsymbol{w}$ is the weight vector and b_1 is the bias term. This output r_{hj} characterizes the preference of u_h to i_j. We denote the preference score from the mirror structure in Figure1 as r'_{hk}. Ultimately, the model prediction is $r_{hj} - r'_{hk}$.

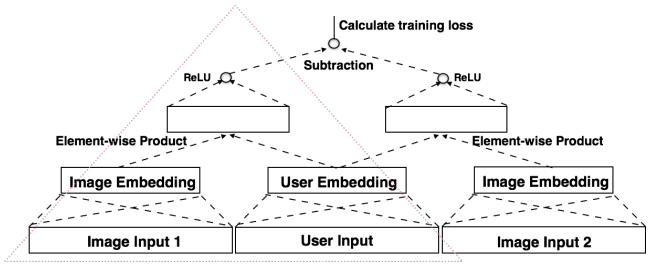

Figure 1: Neural Personalized Ranking (NPR) Structure

4.2 Objective Function

We define the objective function to maximize as:

$$\frac{1}{n} \sum_{\substack{h \in \mathcal{U}, (i_j \in \mathcal{P}_h, i_k \in \mathcal{N}_h \\ | i_j \in \mathcal{N}_h, i_k \in \mathcal{P}_h)}} \ln \left(\delta\big((r_{hj} - r'_{hk}) \cdot g(h,j,k)\big) \right) - \lambda_\Theta \|\Theta\|^2$$

where n is the number of training samples and $\delta(\cdot)$ is the sigmoid function. Since we only focus on whether the sign of the output is the same as $g(h, j, k)$, we employ the product between the predicted value $r_{hj} - r'_{hk}$ and the ground truth $g(h, j, k)$ as an indicator for how the predicted value is aligned with ground truth. A larger value is acquired if their signs are the same. The regularization term is slightly different from that defined in the BPR-based model. We impose the L2-norm to the whole embedding matrix, instead of on each training sample for simpler implementation. If training samples are balanced for each user and image, such regularization will have the same effect as in the BPR model.

4.3 Model Training and Inference

We initialize the weight matrices with random values uniformly distributed in [0,1]. To train the network, we transform the objective to the equivalent minimization dual problem and adopt mini-batch gradient descent (MB-GD), which is a compromise between gradient descent (GD) and stochastic gradient descent (SGD). MB-GD converges faster than GD as it has frequent gradient updates while convergence is more stable than SGD. Besides, MB-GD allows utilization of vectorized operations from deep learning libraries, which typically results in a computational performance gain over SGD. Before each epoch, we shuffle the training dataset. Then in each step, a batch of training tuples is served to the network. The error gradient is back propagated from output to input and parameters in each layer are updated. The batch size we use in experiments is 1,024. The optimization algorithm used for gradient update is Adam's[16]. The loss generally converges within 20 epochs given the amount of training data.

Given a user u, for every image $i \in \mathcal{N}_u$, her preference score r_{ui} is predicted from the neural network. In order to obtain the preference score, we feed the tuple (u, i, i) to the neural network, and get two values r_{ui} and r'_{ui} from the parallel branches. The

final preference score is calculated as $r_{ui} = \frac{1}{2}(r_{ui} + r'_{ui})$. Then the set of images with unobserved feedback are sorted according to descending predicted preference score. We pick the top ranking images for recommendation.

4.4 Implementation Details

Neural network models can easily overfit. Thus we take a few measures to prevent overfitting. First, we apply dropout to the embedding weights during training. The dropout rate is fine-tuned for each dataset. Second, if validation loss does not decrease, we reduce the learning rate to 20% of its current value, allowing for finer adjustment to gradient update. Third, early stopping is adopted to terminate training if there is no decrease on validation loss for 3 epochs. Additionally, we impose L2-regularization to the contextual preference vectors for contextual NPR model, which we will introduce in the following section, such that the preference score is not overwhelmed by large contextual feature values. Furthermore, all regularization coefficients are tuned through grid search.

5 CONTEXTUAL NPR

Although the neural personalized ranking model is promising, it faces two key challenges. The first is *sparsity* – very few images have been liked, so it is difficult to make recommendations for users who have little feedback as well as to recommend newly posted images. The second is *preference complexity* – images are diverse and there are many reasons for a user to like an image. Hence, we propose to improve NPR with an enhanced model – *contextual neural personalized ranking (C-NPR)* – by leveraging multiple categories of auxiliary information that may help overcome the sparsity issue while also providing clues to user preferences.

5.1 Geo, Topical, Visual Preference

Based on the Flickr YFCC100M dataset [33] (see Section 6.1), we begin here by highlighting evidence for the impact of three sources of contextual information on image preference, before formally defining the contextual NPR model.

Evidence of Spatial Preference. Figure 2 shows the percentage distribution of "liked" images in decreasing order across the regions where these images were taken. Here we aggregate each user's top-10 regions where their liked images come from. The k^{th} boxplot is generated from all users who have liked images from at least k regions. We observe that the median percentage of liked images from the top region is above 33%; that is, at least half of all users have 33% of their liked images from a single region (though not necessarily the same region for each user). Suppose a user has no preference of regions, a single region would at most contain 9% of her liked images (as the largest region contains 9% of the images). Thus we conclude there is a strong tendency for a user to favor images from certain regions, especially from a few of them as the percentage decreases sharply as the region number increases.

Evidence of Topic Preference. We consider each unique user tag as a potential topic. Figure 3 shows users' liked image distribution over the tags that have been applied to those images. We list the results for the top-10 tags of each user (not necessarily the same set of tags for each user). The k^{th} boxplot summarizes users that have more than k tags labeled to the set of liked images. We observe that ~75% of users have at least one common tag shared among more than ~35% of their liked images. Even the median ratio for the 10^{th} tag attached to liked images is much higher than the percentage of most frequent tags in the whole dataset. Thus we conclude that users have topic preferences for the images they like. As the percentage decreases slowly with k, we ascribe this to users having multiple favored tags.

Evidence of Visual Preference. Finally, we explore clues for user's visual preference by comparing image similarity across three sampled sets, with each containing 100,000 image pairs. The sets are constructed in the following manner: (i) Randomly sample image pairs; (ii) Randomly sample a user, then sample a pair of image from her liked images; and (iii) For each image, pick its most socially alike images. Here we represent each image as a vector of user's who like it, then identify similar images with high cosine similarity score. Next, we calculate the cosine similarity of the aforementioned image pairs based on their visual feature vectors. The similarity distributions for these three sets are shown in Figure 4. We observe that image pairs liked by a user tend to be more similar in visual appearance than randomly picked image pairs, with a median similarity around 0.25 vs. 0.20. For image pairs that are liked by similar groups of users, the pairwise visual similarity is even higher, reaching 0.30. All three findings are statistically significant with p-value less than 1e-8. Hence, we conclude that users have visual preference for images that they like, and that there exists group of users that share similar preferences.

5.2 From NPR to C-NPR

This evidence of clear variation in user preference motivates our need to augment NPR. Formally, with the contextual feature vector $\bar{\boldsymbol{f}}_{\boldsymbol{i}}$ for image i, we then seek to uncover user's preference latent vector $\boldsymbol{f}_{\boldsymbol{u}}$ to $\bar{\boldsymbol{f}}_{\boldsymbol{i}}$ such that the vector product $\boldsymbol{f}_{\boldsymbol{u}} \circ \bar{\boldsymbol{f}}_{\boldsymbol{i}}$ captures how user preference is aligned with the image's contextual features.

We modify the neural network structure of each branch in Figure 1 to accommodate for modeling contextual preference. The new architecture for a branch incorporating visual, geo, and topic contextual features and preferences is shown in Figure 5. Aside from the user and image input, each category of contextual features of image $\bar{\boldsymbol{v}}_i, \bar{\boldsymbol{t}}_i, \bar{\boldsymbol{g}}_i$ is served as an extra input. Each corresponding contextual preference hidden layer is fully connected above user input and is to be learned. Then the user's preference to the contextual feature is calculated with the element-wise product to measure how features and preferences are aligned. Specifically, the visual, topic, and geo latent vectors of user u_h are calculated as:

$$\boldsymbol{v}_h = \boldsymbol{W}_v \boldsymbol{u}_h, \quad \boldsymbol{t}_h = \boldsymbol{W}_t \boldsymbol{u}_h, \quad \boldsymbol{g}_h = \boldsymbol{W}_g \boldsymbol{u}_h$$

where $\boldsymbol{W}_v, \boldsymbol{W}_t, \boldsymbol{W}_g$ are the weight matrices. The visual, topic, and geo preference of user u_h to image i_j are:

$$\begin{aligned} \boldsymbol{e}^v_{hj} &= \boldsymbol{v}_h \circ \bar{\boldsymbol{v}}_j \\ \boldsymbol{e}^t_{hj} &= \boldsymbol{t}_h \circ \bar{\boldsymbol{t}}_j \\ \boldsymbol{e}^g_{hj} &= \boldsymbol{g}_h \circ \bar{\boldsymbol{g}}_j \end{aligned}$$

Then the general preference $\boldsymbol{m}_{hj}$ and contextual preferences are concatenated in the merge layer. Formally:

$$\boldsymbol{m'}_{hj} = \begin{bmatrix} \boldsymbol{m}_{hj} & \boldsymbol{e}^v_{hj} & \boldsymbol{e}^t_{hj} & \boldsymbol{e}^g_{hj} \end{bmatrix}^T$$

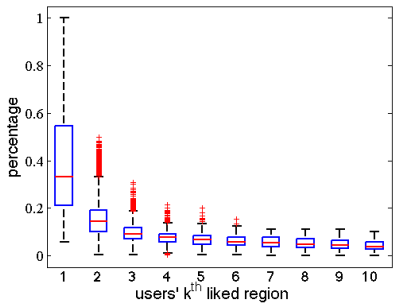

Figure 2: Geo preferences: Users tend to "like" images from only a few regions.

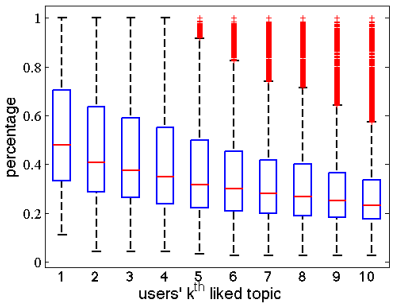

Figure 3: Topic preferences: Users tend to "like" images with similar tags.

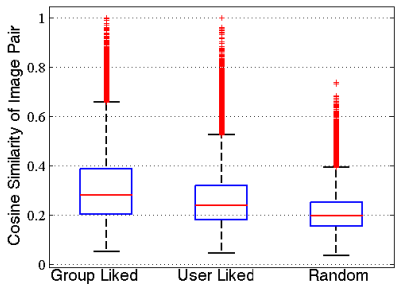

Figure 4: Visual preferences: Pairs of "liked" images tend to be more visually alike than random pairs.

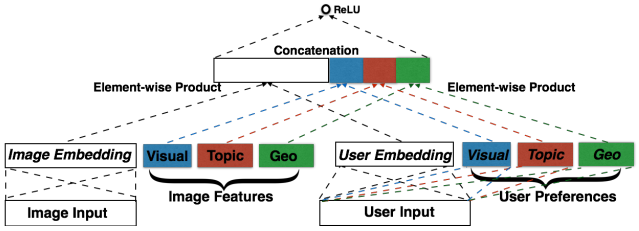

Figure 5: NPR with Contextual Information

Finally, the merge layer is further connected to a single neuron dense layer as before. The updated preference score is:

$$r_{hj} = a(\boldsymbol{w'^T} \boldsymbol{m'}_{hj} + b_1)$$

Additional contextual information about each image can be incorporated following the same steps as stated above. In summary, each new feature vector is served as an extra input to the neural network, and a corresponding preference embedding layer is augmented on top of user input. Then the element-wise product is adopted to model consistency between preferences and intrinsic properties of the image, followed by concatenation of all preference components and a weighted sum.

5.3 Modeling Geo, Topical, and Visual

Given the evidence of user preferences, we turn here to model these features for integration into the C-NPR model.

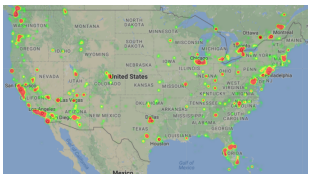

Figure 6: Image Heatmap

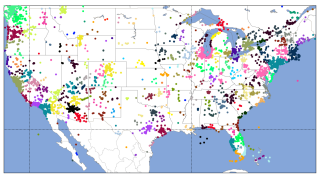

Figure 7: Geographic Regions

Deriving Spatial feature. We assume the area of interest is geographically partitioned into K regions and each image is taken from one of the regions. Instead of gridding into blocks of equal area which has been used previously [21, 28], we propose to partition areas into regions according to image density, where the shape and size of a region doesn't have to be consistent and could be irregular. The reason is images are not distributed homogeneously (generally, dense around cities and tourist attractions and sparse elsewhere). Focusing on density helps to reduce the irrelevant areas and the size of each region we drill down into, which allows for more precise modeling. We apply the mean shift clustering algorithm, which builds upon the concept of kernel density estimation (KDE), to identify geographical clusters of images. It works by placing a Gaussian kernel on each image coordinate in the dataset. Then by iteratively shifting each point in the data set until they reach the top of their nearest KDE surface peak. The only parameter to set is the bandwidth, with which it attempts to generate a reasonable number of clusters based on the density. The clustering result is shown in Figure 7, where each dot represents an image and the cluster of points represents a region. In total, there are 217 regions with a bandwidth of 100km.

We assume the probability that a user likes an image in one region is influenced by her likes status in other regions. If a user has liked images from region p, then she has a larger probability of favoring an image in a region closer to p. Previous work in POI recommendation assumes the influence distance of a POI is fixed according to a normal distribution $\mathcal{N}(0, \sigma^2)$ [21]. However, it is commonly perceived that influence for regions of the same size should be different, not to mention the diverse shape and size in our scenario. Thus we assume each region p has an influence according to a normal distribution $\mathcal{N}(0, \sigma_p^2)$, where σ_p is the standard deviation of distance from each image coordinate to the cluster center. To this end, the influence from p_i to p_j is defined as: $f_{ij} = \frac{1}{\sigma_{p_i}} K(\frac{d(i,j)}{\sigma_{p_i}})$, where p_i and p_j are the regions that image i and j belong to and the relation between image and region is many to one. $d(\cdot)$ is the Haversine distance between the center of two regions, $K(\cdot)$ is the standard normal distribution and σ_{p_i} is the standard deviation which we adopt as the bandwidth of the kernel function. Thus the influence from each region to all other regions is represented as a row vector. The advantage is it encodes the idea of kernel density estimation where the estimated geographical density of u's liked image distribution at p_j is: $d_u^j = \sum_{p_i \in P_u} \frac{n_i}{\sigma_{p_i} |P_u|} K(\frac{d(i,j)}{\sigma_{p_i}})$, where n_i is u's number of likes within p_i, P_u is the set of regions that u has likes. It can be written as the dot product of two vectors. However, different from the KDE, a user's preference vector is learned.

Deriving Topic Features. To extract the topical theme associated with each image, we aggregate the user-generated tags, title, and description (if any) for each image. This text not only acts as a descriptor of concrete objects, scenes, and weather, but also sheds light on abstract and hidden knowledge about the images like emotion and background theme, which supplements the visual appearance. We ignore tags which have occurred fewer than d times in the dataset and apply dimensionality reduction over 58k unique tags.[1]

Deriving Visual Features. Recently, high-level visual features extracted from deep convolutional neural networks have revolutionized the state-of-the-art performance in image recognition [32] and image captioning [38]. Here, the output of fc6 layer of the Places Hybrid-CNN is adopted as the image feature [42], which contains 4,096 dimensions. This CNN was trained on 1,183 categories which includes 205 scene categories from Places Database and 978 object categories from ImageNet (ILSVRC2012) images. Dimension reduction is further applied for reducing computation complexity. The existing approach for visual BPR [9], which learns an embedding kernel for visual dimension reduction while training the recommendation model, turns out to be less efficient than directly utilizing the full set of 4,096 features. Hence, we propose to reduce visual feature dimension separately from model training.[2]

6 EXPERIMENTS

In this section, we conduct a set of experiments to evaluate neural personalized image recommendation. Specifically, we first introduce the data preparation workflow and basic experimental setup. Then we compare NPR with baseline models, followed by reporting performance of contextual enhanced models. We drill down to discover the impact of each category of contextual information. We further look into the performance of the proposed model in the typical cold start scenario. Finally, we discuss the characteristics of NPR and BPR in terms of amount of training data required and convergence rate.

6.1 Data

The dataset we use for evaluation is based on the Flickr YFCC100M dataset [33]. We select images with geo-coordinates and that are located in the US mainland. We further crawl the image "likes" from the Flickr API and we select images with greater than 30 likes overall and users with more than 10 liked images.

The resulting datasets are listed in Table 1, where the sparsity for the small dataset and large dataset is 0.96% and 0.16%, respectively, which means only 0.96% and 0.16% of the possible user-image relations is available. These two datasets represent two different levels of feedback sparsity. And effective sparsity for training data is half of the reported value after train/test split. The geographical heatmap of the large dataset is shown in Figure 6; we notice the majority of images come from populated areas or famous tourist sites, as shown in red.

[1]We compare principal component analysis (PCA) and Latent Dirichlet Allocation (LDA) for carrying out this task. We report PCA-based results due to its better performance.

[2]We compare recommendation performance with reduced feature from PCA and stacked auto-encoder (AE) as well as with full set of features. Both PCA and AE perform similarly and provide a good trade-off between efficiency and accuracy, thus we only report PCA due to the space limit.

Dataset	#Users	#Images	#Feedback	Sparsity
Small	1,891	2,013	36,827	0.96%
Large	27,782	21,720	961,506	0.16%

Table 1: Post-processed Datasets Statistics

6.2 Experimental Setup

All experiments for BPR-based models were performed on a desktop machine with 60GB memory and 8 core Intel i7-4820k3.7GHz. NPR-based models are trained using Nvidia GeForce GTX Titan X GPU with 12 GB memory and 3,072 cores.

Constructing the Training Set. We randomly partition the liked images of each user into 50% for training and validation and 50% for testing. The validation set split ratio is 0.3. The loss on the validation set is used for tracking training progress. The training set consists of tuples (h, j, k) where h, j, k correspond to user index, positive image index, and negative image index, respectively. Including every pair of positive and negative combination for each user in training would be costly. Yet practically, evaluation metrics saturate even with a much smaller set of training tuples. Thus we propose to use a sampling method for generating training tuples.

To generate each training tuple, we first randomly sample a user u from user set $\mathcal{U}$, then randomly sample a positive image i_j from $\mathcal{P}_u$, and finally randomly sample k negative images i_k from $\mathcal{N}_u$ to pair with i_j. We repeat this process until generating the expected number of training data tuples. The influence of k on performance is discussed later; we set k to 10. All reported results in this paper are based on models trained over a set where the number of sampled users equals to five times the number of observed "likes".

Although it is very likely that we end up leaving part of the positive samples unused, the model based on this sampling strategy exhibits better overall performance and requires less training data to converge compared with sampling negatives for each positive sample. The model is trained in a balanced way among every user and not biased towards users that have more likes.

Evaluation Metrics. We adopt precision@k, recall@k and F1-score@k for evaluating personalized ranking. Precision measures the fraction of correctly predicted images among the retrieved images. Recall measures the fraction of relevant images that have been picked over the total relevant images. F1@k is a weighed average of Prec@k and Rec@k. All measures are averaged across all users.

$$Prec@k = \frac{1}{N}\sum_{i=1}^{N}\frac{|GT(u_i) \cap Pred(u_i)@k|}{k}$$

where $GT(u_i)$ is the ground truth liked image for u_i in test data, and $Pred(u_i)@k$ is the top k recommended images for u_i.

$$Rec@k = \frac{1}{N}\sum_{i=1}^{N}\frac{|GT(u_i) \cap Pred(u_i)@k|}{|GT(u_i)|}$$

$$F1@k = \frac{2 \cdot Prec@k \cdot Rec@k}{Prec@k + Rec@k}$$

Baselines.

- NCF. Neural collaborative filtering is a pointwise model composed of multi-layer perceptron and generalized matrix factorization components [10]. All the configurations adopted are similar according to original paper including 4 hidden layers, 64 hidden units and pre-training. We sample 5 negative examples for each positive, which was shown to be optimal in the original paper.

- Multi-layer perceptron based pairwise ranking model. A personalized pairwise ranking model based on multi-layer perceptron was introduced [34]. We adopt a setting with 3 hidden layers, with each layer containing 200, 100, and 100 units, respectively.
- BPR and its variants. We consider the basic pairwise ranking for matrix factorization model shown in Equation (1). In addition, we can also integrate the proposed contextual factors into traditional BPR. Indeed, a visual preference-enhanced version of BPR model has been previously introduced by He et al.[9]. Hence, we also consider a visual (VBPR), topic (TBPR), geo (GBPR), and combined version of BPR (C-BPR).
- NPR. This is the neural network based model for personalized pairwise ranking as shown in Figure 1.
- NPR-noact. This is the NPR model without nonlinear activation.
- Contextual NPR. This includes NPR considering only visual (VNPR), topic (TNPR), and geo (GNPR) contextual information.

Reproducibility. For all models, the user and image latent factor dimensions are set to 100 empirically for a trade-off between performance and computation complexity as well as for fair comparison. The number of visual feature dimensions is 128, the number of topic dimensions is 100 for the small dataset and 500 for the large dataset. The number of geographic dimensions is the same as the number of geo clusters which is 155 and 217 for small and large dataset, respectively.

For the NPR-based approach, we adopt mini-batch gradient descent where the batch size is set to 1,024. The dropout rate for the small dataset was set to 0.6 and for the large dataset was set to 0.45. The regularization parameters are fine-tuned. For example, on the large dataset λ_u=λ_i=1e−7, λ_v=λ_g=1e−6, and λ_t=1e−5. For BPR-based approaches, we initialize the learning rate to 0.02 and decrease it to 97% its current value in each consecutive iteration, which has been shown to be effective to help convergence in fewer iterations [12]. And generally, training converges within 80 iterations. The regularization parameters are fine-tuned and shared among all BPR baselines, concretely, λ_u=λ_i=λ_b=0.02, λ_v=λ_g=0.01 and λ_t=0.1.

6.3 NPR vs. Alternatives

We begin by investigating the quality of NPR versus each of the baselines for personalized recommendation without contextual information. We report the average precision@k, recall@k for k at 5, 10, 15 in Figure 8 for the small dataset and 9 for the large dataset. We observe that NPR and BPR are neck and neck, with BPR slightly superior (less than 1%) in precision and recall. This indicates BPR-MF is a strong baseline. Although the MF component is linear, the logistic objective function brings in nonlinearity. Both approaches consistently substantially outperform other baseline approaches in precision and recall. Moreover, the pairwise method generally yields better results. For example, NPR improves the precision and recall over the pointwise model NCF by 50% for the large dataset and improves the precision and recall. This illustrates the relaxed assumption for unobserved samples helps to reduce the recommendation bias. The nonlinear activation function lead to an average of 3.8% increase in precision and 3.3% increase in recall on the small dataset, and even larger 11.5% and 12.5% increase in precision and recall on the large dataset. By bringing in nonlinearity, the representativeness of the model is enriched. We observe the performance metrics are generally lower on the large dataset; the reason is that recommendation becomes more difficult given more images and increasing sparsity. However, the performance gap between approaches expands with increasing sparsity, indicating the great opportunity for the proposed approach when feedback is lacking.

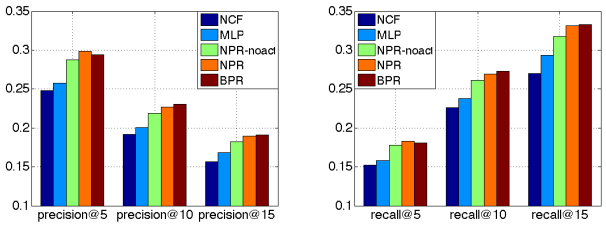

Figure 8: Average Precision and Recall for Baseline Models on the Small Dataset

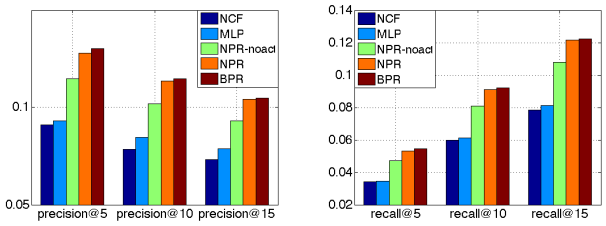

Figure 9: Average Precision and Recall for Baseline Models on the Large Dataset

6.4 Comparing Contextual Enhanced Models

To evaluate the impact of incorporating each category of contextual information in recommendation, we present precision and recall at k for each contextual enhanced NPR and BPR model over the large dataset in Tables 2 and 3. We observe that modeling additional contextual factors improves over the basic NPR and BPR method. Concretely, TNPR gives an average improvement of 10.1% in precision and 12.6% in recall over the NPR baseline on the large dataset. This indicates that rich textual side knowledge acts as an effective filter for sifting relevant images. VNPR performs slightly better than NPR, with an improvement of 4.6% and 5.4% in precision and recall. The lesson here is learning personal visual preference does help to connect users with images that have appearance agreement. Furthermore, GNPR gives an average of 3.6% percent and 5.5% percent increase in precision and recall. This confirms the importance of modeling user's geographical region which is consistent with our observation in Section 5.1, where we notice the user's strong geographical preference. Since for social image sharing sites, users do have connections focusing around their home location and places they are familiar with, we see that images in these regions may be more likely to be related with the user. Finally, we observe that C-NPR achieves an average of more than 16% increase in precision and recall. This implies that the proposed model is effective in integrating various categories of contextual information jointly to

make better recommendation. We observe similar trends in C-BPR models.

Method	p@5	p@10	avg Δ	r@5	r@10	avg Δ
NPR	0.1280	0.1137	-	0.0531	0.0909	-
VNPR	0.1354	0.1177	+4.6%	0.0563	0.0952	+5.4%
TNPR	0.1411	0.1250	+10.1%	0.0599	0.1021	+12.6%
GNPR	0.1326	0.1178	+3.6%	0.0564	0.0953	+5.5%
C-NPR	0.1504	0.1317	+16.6%	0.0644	0.1081	+16.6%

Table 2: Integrating Contextual Information in NPR

Method	p@5	p@10	avg Δ	r@5	r@10	avg Δ
BPR	0.1302	0.1148	-	0.0544	0.0920	-
VBPR	0.1366	0.1188	+4.2%	0.0577	0.0961	+5.3%
TBPR	0.1384	0.1217	+8.5%	0.0588	0.0992	+8.0%
GBPR	0.1331	0.1171	+2.1%	0.0562	0.0950	+3.3%
C-BPR	0.1445	0.1255	+10.6%	0.0619	0.1034	+13.1%

Table 3: Integrating Contextual Information in BPR

6.5 NPR and BPR with Contextual Information

First, even though the NPR base model performs similarly with BPR, we observe that C-NPR leads by an average of ~4.5% higher precision and recall over C-BPR on the large dataset and ~1.5% increase on the small dataset. We ascribe this improvement to the neural network based model flexibly adjusting weights for each feature dimension and nonlinear activation enriching the expressiveness. Second, the higher increase for the large dataset indicates the C-NPR model could be more beneficial than the C-BPR model for recommendation under the real-world scenario of extreme feedback sparsity.

Method	p@5	p@10	r@15	r@5	r@10	r@15
C-NPR(S)	0.2987	0.2371	0.1977	0.1866	0.2842	0.3471
C-BPR(S)	0.3034	0.2335	0.1945	0.1874	0.2801	0.3419
C-NPR(L)	0.1504	0.1317	0.1192	0.0644	0.1081	0.1430
C-BPR(L)	0.1445	0.1255	0.1141	0.0619	0.1034	0.1363

Table 4: Compare Contextual NPR and Contextual BPR

6.6 Cold Start

In this experiment, we focus on the cold-start scenario which is commonly encountered in recommendation where we have a limited number of positive user feedbacks for training the model. Here we select users who have fewer than seven liked images to examine the performance of the proposed model on the large dataset in the cold-start setting. Interestingly we observe in Table 5 that the proposed C-NPR model outperforms the baseline NPR model by average ~21% in precision and recall. Additionally, each contextual model exhibits better performance than the NPR baseline, with TNPR taking the lead showing an average improvement of ~13% in precision and recall. This implies these contextual factors help to alleviate the sparsity in the cold-start setting. Moreover, the lager improvement compared with ordinary setting again validates our claim that contextual information is especially helpful when feedback is rare.

Method	p@5	p@10	p@15	r@5	r@10	r@15
NPR	0.0723	0.0598	0.0518	0.0643	0.1063	0.1381
VNPR	0.0775	0.0628	0.0554	0.0683	0.1131	0.1455
TNPR	0.0820	0.0678	0.0584	0.0731	0.1206	0.1558
GNPR	0.0775	0.0644	0.0563	0.0685	0.1120	0.1455
C-NPR	0.0893	0.0721	0.0626	0.0769	0.1282	0.1668

Table 5: NPR Cold-start Performance

6.7 Number of training samples

In this experiment, we explore how performance of different models is influenced by the amount of training data used as well as by the number of negative samples for each positive one. As mentioned in Section 6.2, the training data generation procedure is as follows: for NPR-1 and BPR-1, we first randomly sample a user, then sample one positive (liked) image for the user and followed by one negative (unobserved/ disliked) image of the user. For NPR-10 and BPR-10 instead, we randomly sample ten negative image for each positive image while keeping other steps the same. The total number of training tuples generated is measured in terms of the number of positive feedbacks in the original dataset. In Figure 10, the horizontal axis represents the number of times (of positive feedback) to sample and the vertical axis is the F-1 score@10. We observe that BPR-1 and NPR-1 achieve increasing F-1 score with gradual increase in training data. However, the performance of NPR-1 and BPR-1 models have disparate properties. First, the increase for the NPR-based model is relatively gentle, while steeper for the BPR-based model. Furthermore, the NPR model performs much better, for example, it gains 0.23 for F-1 score@10 at 5 times of sampling while the BPR-based model only reaches 0.17. The difference in performance is more severe when training data is lacking. Interestingly, we also notice that the neural network based model generally achieves better performance with inadequate training data. We attribute this to the linear model having less powerful expressiveness, hence incurring overfitting more easily and vice versa for nonlinear models. The performance gap doesn't decrease even after we adjust the regularization parameters to their optimal setting. To note, the same phenomenon was observed on the large dataset. After we decoupled the negative samples sampled for training, we notice better F1 score for both approaches, yet the performance curve gradually saturates as we continue serving more training data. This indicates the models would stop improving as the size of training data is no longer the bottleneck. To our best knowledge, this is the first effort to compare such differences in behavior of these two categories of models, and we hope this observation will provide some reference for further research.

7 CONCLUSION

In this paper, we tackle the problem of personalized image recommendation. We propose *Neural Personalized Ranking (NPR)* – a new neural network based personalized pairwise ranking model for implicit feedback, which incorporates the idea of generalized matrix factorization. We further build an enhanced model by augmenting the basic NPR model with users' multiple contextual preference clues and derive corresponding features that can be incorporated into both the NPR and the BPR frameworks to better uncover user

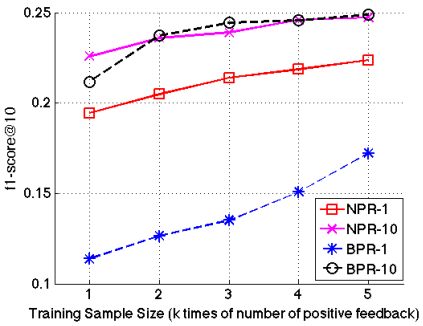

Figure 10: Performance w.r.t Training Sample Size

preferences. Through extensive experimental validation, we demonstrate the proposed NPR model significantly outperforms several state-of-the-art approaches. Moreover, we observe the superiority of contextual enhanced NPR model over the baseline model.

In future work, we are interested to incorporate user information like demographics into the framework for improving the quality of recommendation, especially for new users. Additionally, we would like to extend the current model with additional contextual information, for example, modeling the temporal evolution of preferences by revising certain model components with LSTM. Furthermore, we are eager to develop a distributed model for large scale recommendation.

Acknowledgments This work was supported in part by NSF grant IIS-1149383.

REFERENCES

[1] Gediminas Adomavicius and Alexander Tuzhilin. 2011. Context-aware recommender systems. In *Recommender systems handbook*. Springer.
[2] David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research* (2003).
[3] Anders Brodersen, Salvatore Scellato, and Mirjam Wattenhofer. 2012. Youtube around the world: geographic popularity of videos. In *WWW*. ACM.
[4] Chen Cheng, Haiqin Yang, Irwin King, and Michael R Lyu. 2012. Fused Matrix Factorization with Geographical and Social Influence in Location-Based Social Networks.. In *AAAI*.
[5] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, et al. 2016. Wide & deep learning for recommender systems. In *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems*. ACM.
[6] Paul Covington, Jay Adams, and Emre Sargin. 2016. Deep neural networks for youtube recommendations. In *RecSys*. ACM.
[7] Ali Mamdouh Elkahky, Yang Song, and Xiaodong He. 2015. A multi-view deep learning approach for cross domain user modeling in recommendation systems. In *WWW*. ACM.
[8] Jianping Fan, Daniel A Keim, Yuli Gao, Hangzai Luo, and Zongmin Li. 2009. JustClick: Personalized image recommendation via exploratory search from large-scale Flickr images. *IEEE Transactions on Circuits and Systems for Video Technology* (2009).
[9] Ruining He and Julian McAuley. 2016. VBPR: visual bayesian personalized ranking from implicit feedback. In *AAAI*.
[10] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In *WWW*. ACM.
[11] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. 2015. Session-based recommendations with recurrent neural networks. *arXiv preprint arXiv:1511.06939* (2015).
[12] Longke Hu, Aixin Sun, and Yong Liu. 2014. Your neighbors affect your ratings: on geographical neighborhood influence to rating prediction. In *SIGIR*. ACM.
[13] Yifan Hu, Yehuda Koren, and Chris Volinsky. 2008. Collaborative filtering for implicit feedback datasets. In *ICDM*. IEEE.
[14] Mohsen Jamali and Martin Ester. 2010. A matrix factorization technique with trust propagation for recommendation in social networks. In *RecSys*. ACM.
[15] Yuchen Jing, Xiuzhen Zhang, Lifang Wu, Jinqiao Wang, Zemeng Feng, and Dan Wang. 2014. Recommendation on Flickr by combining community user ratings and item importance. In *ICME*.
[16] Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).
[17] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer* (2009).
[18] Chenyi Lei, Dong Liu, Weiping Li, Zheng-Jun Zha, and Houqiang Li. 2016. Comparative Deep Learning of Hybrid Representations for Image Recommendations. In *CVPR*. IEEE.
[19] Yuncheng Li, Jiebo Luo, and Tao Mei. 2014. Personalized image recommendation for web search engine users. In *Multimedia and Expo (ICME), 2014 IEEE International Conference on*. IEEE.
[20] Yuncheng Li, Tao Mei, Yang Cong, and Jiebo Luo. 2015. User-curated image collections: Modeling and recommendation. In *IEEE International Conference on Big Data*.
[21] Defu Lian, Cong Zhao, Xing Xie, Guangzhong Sun, Enhong Chen, and Yong Rui. 2014. GeoMF: joint geographical modeling and matrix factorization for point-of-interest recommendation. In *SIGKDD*. ACM.
[22] Bin Liu, Yanjie Fu, Zijun Yao, and Hui Xiong. 2013. Learning geographical preferences for point-of-interest recommendation. In *SIGKDD*. ACM.
[23] Shaowei Liu, Peng Cui, Wenwu Zhu, Shiqiang Yang, and Qi Tian. 2014. Social embedding image distance learning. In *MM*. ACM.
[24] Xianming Liu, Min-Hsuan Tsai, and Thomas Huang. 2016. Analyzing User Preference for Social Image Recommendation. *arXiv preprint arXiv:1604.07044* (2016).
[25] Hao Ma, Dengyong Zhou, Chao Liu, Michael R Lyu, and Irwin King. 2011. Recommender systems with social regularization. In *WSDM*. ACM.
[26] Augusto Q Macedo, Leandro B Marinho, and Rodrygo LT Santos. 2015. Context-aware event recommendation in event-based social networks. In *RecSys*.
[27] Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton Van Den Hengel. 2015. Image-based recommendations on styles and substitutes. In *SIGIR*. ACM.
[28] Wei Niu, James Caverlee, Haokai Lu, and Krishna Kamath. 2016. Community-based geospatial tag estimation. In *ASONAM*.
[29] Rong Pan, Yunhong Zhou, Bin Cao, Nathan N Liu, Rajan Lukose, Martin Scholz, and Qiang Yang. 2008. One-class collaborative filtering. In *Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on*. IEEE.
[30] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian personalized ranking from implicit feedback. In *Proceedings of the twenty-fifth conference on uncertainty in artificial intelligence*. AUAI Press.
[31] Jitao Sang and Changsheng Xu. 2012. Right buddy makes the difference: An early exploration of social relation analysis in multimedia applications. In *MM*. ACM.
[32] K. Simonyan and A. Zisserman. 2014. Very Deep Convolutional Networks for Large-Scale Image Recognition. *CoRR* abs/1409.1556 (2014).
[33] Bart Thomee, David A Shamma, Gerald Friedland, Benjamin Elizalde, Karl Ni, Douglas Poland, Damian Borth, and Li-Jia Li. 2016. Yfcc100m: The new data in multimedia research. *Commun. ACM* (2016).
[34] Mikhail Trofimov, Sumit Sidana, Oleh Horodnitskii, Charlotte Laclau, Yury Maximov, and Massih-Reza Amini. 2017. Representation Learning and Pairwise Ranking for Implicit and Explicit Feedback in Recommendation Systems. *arXiv preprint arXiv:1705.00105* (2017).
[35] Aaron Van den Oord, Sander Dieleman, and Benjamin Schrauwen. 2013. Deep content-based music recommendation. In *Advances in neural information processing systems*.
[36] Hao Wang, Naiyan Wang, and Dit-Yan Yeung. 2015. Collaborative deep learning for recommender systems. In *SIGKDD*. ACM.
[37] Yao Wu, Christopher DuBois, Alice X Zheng, and Martin Ester. 2016. Collaborative denoising auto-encoders for top-n recommender systems. In *WSDM*. ACM.
[38] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *ICML*.
[39] Mao Ye, Peifeng Yin, Wang-Chien Lee, and Dik-Lun Lee. 2011. Exploiting geographical influence for collaborative point-of-interest recommendation. In *SIGIR*. ACM.
[40] Fuzheng Zhang, Nicholas Jing Yuan, Defu Lian, Xing Xie, and Wei-Ying Ma. 2016. Collaborative knowledge base embedding for recommender systems. In *SIGKDD*.
[41] Jia-Dong Zhang and Chi-Yin Chow. 2015. GeoSoCa: Exploiting geographical, social and categorical correlations for point-of-interest recommendations. In *SIGIR*. ACM.
[42] Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. 2014. Learning deep features for scene recognition using places database. In *Advances in neural information processing systems*. 487–495.