Do "Also-Viewed" Products Help User Rating Prediction?

Chanyoung Park[1], Donghyun Kim[1], Jinoh Oh[2] and Hwanjo Yu[1]*
[1]Dept. of Computer Science and Engineering, POSTECH, Pohang, South Korea
[2]School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, USA
[1]{pcy1302, kdh5377, hwanjoyu}@postech.ac.kr, [2]jinoho@cs.cmu.edu

ABSTRACT

For online product recommendation engines, learning high-quality product embedding that captures various aspects of the product is critical to improving the accuracy of user rating prediction. In recent research, in conjunction with user feedback, the appearance of a product as side information has been shown to be helpful for learning product embedding. However, since a product has a variety of aspects such as functionality and specifications, taking into account only its appearance as side information does not suffice to accurately learn its embedding. In this paper, we propose a matrix co-factorization method that leverages information hidden in the so-called *"also-viewed"* products, i.e., a list of products that has also been viewed by users who have viewed a target product. *"Also-viewed"* products reflect various aspects of a given product that have been overlooked by visually-aware recommendation methods proposed in past research. Experiments on multiple real-world datasets demonstrate that our proposed method outperforms state-of-the-art baselines in terms of user rating prediction. We also perform classification on the product embedding learned by our method, and compare it with a state-of-the-art baseline to demonstrate the superiority of our method in generating high-quality product embedding that better represents the product.

Keywords

Collaborative filtering; Product embedding; Online shopping

1. INTRODUCTION

Triggered by the Netflix Prize [4] in 2009 whose goal was to predict user ratings on movies based on previous user feedback, the vast majority of research on recommender systems [6] has focused on accurately predicting user ratings [11]. To this end, it is crucial to learn high-quality product embedding, the dimensions of which align with those of user preference, since high-quality product embedding yields higher accuracy in user rating prediction [3]. Therefore, researchers have striven to learn high-quality product embedding by incorporating various side information related to products such as product reviews [10, 25, 26] and product images [8, 9], which help to learn better product embedding, and eventually lead to a better user rating prediction accuracy.

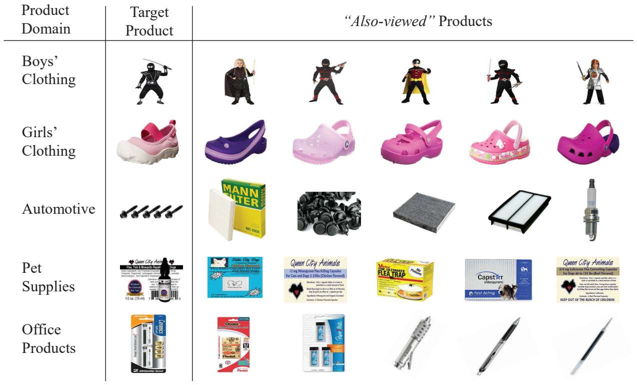

Figure 1: A target product and its *"also-viewed"* products.

In this paper, we study the varying importance of product aspects for users in different product domains. Consider a real-world online shopping scenario where users rate products. What makes users assign high ratings to certain products? That is, what aspects of a product influence user ratings? The products' inherent aspects such as *appearance, functionality* or *specifications* certainly have a vital effect on user ratings. *However, the extent to which each aspect influences users varies among product domains.* For example, in clothing, the appearance of clothes is undoubtedly the most influential factor, whereas in "Office Products", such aspects as functionality and conformance to specifications mainly influence user ratings.

Notably, such phenomena are reflected in online shopping browsing histories in the form of *"also-viewed"* products, i.e., a list of products that have also been viewed by users who have viewed a target product. *"Also-viewed"* product information can be obtained from browsing histories of users. As an intuitive example, consider **Example 1**.

Example 1. Figure 1 shows examples of *"also-viewed"* products in different product domains in *Amazon*[1]. Interestingly, while the *"also-viewed"* products in clothing

*Corresponding author

[1] http://www.amazon.com

WWW'17 , April 3–7, 2017, Perth, Australia.
ACM 978-1-4503-4913-0/17/04.
http://dx.doi.org/10.1145/3038912.3052581

domain (Boys' and Girls' Clothing) look similar, those in other domains (Automotive, Pet supplies, Office products) are not similar in appearance, but are functionally related. Consider "Pet supplies" as an example. Given a liquid flea repellent as a target product, the *"also-viewed"* products include visually different but functionally related products, e.g., flea killing capsules and flea traps.

Example 1 shows that when users shop online, they pay more attention to different aspects of products in different product domains.

Although recent works [8, 9] have successfully taken into account the appearance of products in visually-aware product domain, they have yielded only a slight improvement in other domains where aspects such as functionality and specifications are significant for user ratings. This is because while every product domain has different aspects that are more influential to user ratings as shown in **Example 1**, existing methods only consider product appearance as side information that plays an important role only in clothing domain. Moreover, even in clothing, the appearance of a product is not the only factor that influences user ratings, and thus there is room for further improvement in modeling user ratings. Furthermore, due to the inherent data sparsity, existing methods have modeled users' visual preferences only based on the images of products rated by them in the past, which are usually very few in number. Consequently, the lack of rated products of users gives rise to the insufficiently modeled visual preferences of the users, which eventually degrades the accuracy of user rating prediction.

To address the aforementioned limitations of the existing works in the area, we propose a matrix co-factorization method called Visual Matrix Co-Factorization (VMCF). Our method leverages *"also-viewed"* products that reflect various aspect of a given product that have been previously overlooked by [8, 9]. Precisely, *"also-viewed"* products help systems to learn more high-quality product embedding for two reasons. First, *"also-viewed"* products encode not only visual similarity, but also functional or specification-related similarity, as shown in **Example 1**. Thus, we can capture aspects overlooked by visual features of a product by leveraging *"also-viewed"* products, even in clothing domain. In other domains, where the appearance of a product is not as significant, the explicit relationships among products expressed through *"also-viewed"* information are even more helpful than visual features in building more high-quality product embedding. Second, since most products have an insufficient number of ratings, the explicit relationships between each rated product and its unrated *"also-viewed"* products help us reflect various aspects of products in the product embedding, and thus eventually compensate for a lack of rated products. Moreover, *"also-viewed"* products are more helpful when only a few rated products are given, i.e., *"Cold-start"* setting.

Our main contributions are summarized as follows:

1. To reflect the relationships among products in terms of various aspects such as appearances, functionality and specifications, we build a so-called *product-affinity network* using *"also-viewed"* products.
2. We then simultaneously factorize user ratings data and the *product-affinity network* by sharing the product embedding, which results in high-quality product embedding, and eventually, more accurate user rating prediction.
3. Experimental results on multiple real-world datasets demonstrate that our proposed method significantly outperforms state-of-the-art methods, especially in *"Cold-start"* setting where each product has only a few ratings.
4. By additionally performing classification on product embedding, we empirically demonstrate that the product embedding generated by our method represents the latent dimensions of products better than a state-of-the-art method.

The remainder of this paper is organized as follows: We briefly review related work in Section 2, and provide the formulation of the problem that we solve in Section 3. We then introduce our proposed method in Section 4, followed by results of comprehensive experiments on real-world datasets in Section 5. Finally, the conclusion is presented in Section 6.

2. RELATED WORK

While recommender systems have lately generated a vast amount of research literature, we only review the studies closely related to ours, i.e., matrix factorization and visually-aware approaches for recommendation.

Matrix Factorization.

The goal of matrix factorization (MF) is, given n users and m products, to decompose rating matrix $\mathcal{R} \in \mathbb{R}^{n\times m}$ into two low rank-K matrices $U \in \mathbb{R}^{K\times n}$ and $V \in \mathbb{R}^{K\times m}$, and minimize the reconstruction error as follows [23]:

$$\min_{U,V} \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{m} I_{ij}\left(r_{ij} - U_i^T V_j\right)^2 + \frac{\lambda_u}{2}||U||_F^2 + \frac{\lambda_v}{2}||V||_F^2 \quad (1)$$

where r_{ij} is the rating assigned by user i to product j, $U_i \in \mathbb{R}^K$ and $V_j \in \mathbb{R}^K$ represent the embedding for user i and product j, respectively, where K is the dimensionality of the rating embedding space. I_{ij} equals to 1 if $r_{ij} = 1$, and 0 otherwise. λ_u and λ_v are regularization parameters for user embedding matrix U and product embedding matrix V, respectively, and $||\cdot||_F^2$ denotes the Frobenius norm. The gradient descent-based optimization technique is generally applied to find the local minimum solution for Eq. 1. Matrix factorization is commonly used as a building block for extending recommender systems to incorporate side information such as social networks [7, 15, 19, 20], textual data [1, 10, 25, 26], or both [5, 21], temporal dynamics [12, 27], and product images [8, 9].

Visually-Aware Approaches for Recommendation.

The appearance of a product is one of the most important aspects that impacts user ratings in online shopping. However, it has been commonly ignored because past image feature extraction methods failed to achieve satisfactory performance in visual machine learning tasks such as image classification and object detection. However, recent breakthroughs in deep learning has facilitated such tasks to attain high performance, and thus product image is now recognized as a valuable source of side information for recommender systems. Specifically, image features extracted from a pre-trained Convolutional Neural Network (CNN) effectively represent the latent properties of images. He *et al.* [8, 9] recently introduce visually-aware recommender systems based on the framework of matrix factorization. They embed high-dimensional image features extracted from a pre-trained CNN into low-dimensional features, and use them to model user and product visual embeddings. Their main concern is the appearance of a product, which plays a significant role in clothing domain. However, other product aspects are ignored, such as functionality and specification, which are certainly more significant than appearance in domains such as Automotive, Pet supplies, and Office products.

Meanwhile, McAuley *et al.* [17, 18] use the *"also-viewed"* product information to recommend visually alternative products in clothing domain as a *link prediction* task. Our proposed method is different from this method in that we leverage *"also-viewed"* product information for a *user rating prediction task* to consider *various aspects beyond appearance* in product embedding, which improves user rating prediction *in general domains* not limited to the clothing domain.

3. PROBLEM FORMULATION

3.1 Notations

We first introduce the notations used in this paper. Let $\mathcal{U} = \{u_1, u_2, ..., u_n\}$ be the set of users and $\mathcal{V} = \{v_1, v_2, ..., v_m\}$ be the set of products, where n and m are the number of users and products, respectively. The ratings assigned by users in $\mathcal{U}$ to products in $\mathcal{V}$ are represented by rating matrix $\mathcal{R} = [r_{ij}]_{n\times m}$, where r_{ij} denotes the rating that user i assigns to product j. Depending on the application, r_{ij} can be either a real number or a binary value. When users explicitly express their opinions on products, r_{ij} is a real number, often in the range [1,5], and when $\mathcal{R}$ reflects users' action such as click or non-click and bookmarked or not bookmarked, r_{ij} is a binary value. Although this paper focuses on the former case, it can readily be applied to the latter as well. Without loss of generality, we convert the ratings of 1...5 into the interval [0,1] through normalization. The notations used in the paper are summarized in Table 1.

3.2 Extracting Visual Features

As in [8, 9], we use a pre-trained CNN to extract features from product images. Precisely, we use the CNN architecture proposed by [13], which was trained on 1.2 million ImageNet (ILSVRC2010) images [22]. We pass m products through the pre-trained CNN, and extract the outputs of the second fully-connected layer to construct a product feature matrix $\mathcal{F} \in \mathbb{R}^{C\times m}$, the column vector $f_j \in \mathbb{R}^C$ of which denotes the visual feature vector of product $j \in \mathcal{V}$, where $C = 4096$. Moreover, we introduce an embedding kernel matrix $\mathbf{E} \in \mathbb{R}^{D\times C}$ to transform high-dimensional visual features $f_j \in \mathbb{R}^C$ into D-dimensional product visual embedding space by $\mathbf{E}f_j$.

3.3 Constructing Product-affinity Network

In order to incorporate *"also-viewed"* product information into our method, we build a so-called *product-affinity network* whose nodes denote products and edges encode the *"also-viewed"* relationships among products. As shown earlier in **Example 1**, neighboring products in the *product-affinity network* share common product aspects such as appearance, functionality and specifications, which implies that various aspects of the product are reflected in the network. Note that the edges are directed from a target product to its *"also-viewed"* products, and therefore the relationships are usually not symmetric. We represent the *product-affinity network* as a *product-affinity matrix* $\mathcal{S} = [s_{jk}]_{m\times m}$ such that each entry s_{jk} is defined as:

$$s_{jk} = \begin{cases} 1 & k \in N_j. \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

where N_j denotes the set of *"also-viewed"* products of product j.

Table 1: Notations.

Notation	Explanation
$\mathcal{U}, \mathcal{V}$	User set ($\lvert\mathcal{U}\rvert = n$), Product set($\lvert\mathcal{V}\rvert = m$)
$\mathcal{R}$	Rating matrix ($n \times m$)
$\mathcal{S}$	*Product-affinity matrix* ($m \times m$)
$\mathcal{F}$	Visual feature matrix ($C \times m$)
f_j	Visual feature of product j ($C \times 1$)
r_{ij}	Rating assigned by user i to product j
s_{jk}	$s_{jk} = 1$ if product j and k are neighboring nodes
K	Num. dimensions of product embedding
D	Num. dimensions of product visual embedding
C	Num. dimensions of CNN features
U_i, V_j	Embedding for user i and product j ($K \times 1$)
P_i, Q_j	Visual embedding for user i and product j ($D \times 1$)
Z_k	*"Also-viewed"* product embedding for product k ($K \times 1$)
$\mathbf{E}$	Embedding kernel matrix ($D \times C$)
$g(\cdot)$	Sigmoid function

Given the aforementioned notations, visual features and *product-affinity matrix*, our problem is defined as:

Problem Definition.
Given: *The observed rating matrix $\mathcal{R}$, product visual feature matrix $\mathcal{F}$* and *product-affinity matrix $\mathcal{S}$*
Goal: *Predict missing ratings $r_{ij} \in \mathcal{R}$, where $i \in \mathcal{U}$ and $j \in \mathcal{V}$*

4. METHOD

In this section, we describe our Visual Matrix Co-Factorization (VMCF) method that leverages *"also-viewed"* product information projected on *product-affinity matrix $\mathcal{S}$* in order to take into account various product aspects overlooked by past visually-aware recommendation methods. We first explain how ratings $\mathcal{R}$ and the *product-affinity matrix $\mathcal{S}$* are modeled independently, and demonstrate how these two models are jointly combined using the graphical model described in Figure 2.

4.1 Modeling Rating

Given user-product rating matrix $\mathcal{R} = [r_{ij}]_{n\times m}$, with n users and m products, let r_{ij} represent the rating of user i for product j, and $U \in \mathbb{R}^{K\times n}$ and $V \in \mathbb{R}^{K\times m}$ be user and product embedding matrices, with column vectors U_i and V_j representing user-specific and product-specific embedding vectors, respectively. Moreover, in order to incorporate the visual factors into our model, given user and product visual embedding matrices $P \in \mathcal{R}^{D\times n}$ and $Q \in \mathcal{R}^{D\times m}$ respectively, the term $P_i^T Q_j$ is added [9] where $P_i \in \mathbb{R}^D$ and $Q_j \in \mathbb{R}^D$ denote the user-specific and product-specific visual embedding vectors, respectively, whose inner product models the visual correspondence between user i and product j. Given the embedded visual feature $Q_j = \mathbf{E}f_j$ as explained in Section 3.2, we define the conditional distribution over the observed ratings as:

$$P(\mathcal{R}|U,V,P,\mathbf{E},\sigma_R^2) = \prod_{i=1}^{n}\prod_{j=1}^{m}\Big[\mathcal{N}(r_{ij}|g(U_i^T V_j + P_i^T Q_j),\sigma_R^2)\Big]^{I_{ij}^R}$$

$$= \prod_{i=1}^{n}\prod_{j=1}^{m}\Big[\mathcal{N}(r_{ij}|g(U_i^T V_j + P_i^T \mathbf{E} f_j),\sigma_R^2)\Big]^{I_{ij}^R} \tag{3}$$

where $\mathcal{N}(x|\mu,\sigma^2)$ denotes the probability density function of a Gaussian distribution with mean μ and variance σ^2, and I_{ij}^R is the indicator function that is equal to 1 if user i rated product j, and 0 otherwise. We use the logistic function $g(\cdot)$ to restrict the range of $U_i^T V_j + P_i^T \mathbf{E} f_j$ within [0,1], and convert rating r_{ij} into the range [0,1]. For each hidden variable, we place zero-mean spherical Gaussian priors [23] as follows:

$$P(U|\sigma_U^2) = \prod_{i=1}^{n}\mathcal{N}(U_i|0,\sigma_U^2 I),\ P(V|\sigma_V^2) = \prod_{j=1}^{m}\mathcal{N}(V_j|0,\sigma_V^2 I)$$

$$P(P|\sigma_P^2) = \prod_{i=1}^{n}\mathcal{N}(P_i|0,\sigma_P^2 I),\ P(\mathbf{E}|\sigma_{\mathbf{E}}^2) = \prod_{p=1}^{D}\prod_{q=1}^{C}\mathcal{N}(\mathbf{E}_{pq}|0,\sigma_{\mathbf{E}}^2) \tag{4}$$

Given Eqs. 3 and 4, we can compute the log-posterior distribution over the hidden variables:

$$\begin{aligned} &P(U,V,P,\mathbf{E}|\mathcal{R},\sigma_U^2,\sigma_V^2,\sigma_P^2,\sigma_{\mathbf{E}}^2,\sigma_R^2) \\ &\propto P(\mathcal{R}|U,V,P,\mathbf{E},\sigma_R^2)P(U|\sigma_U^2)P(V|\sigma_V^2)P(P|\sigma_P^2)P(\mathbf{E}|\sigma_{\mathbf{E}}^2) \\ &= \prod_{i=1}^{n}\prod_{j=1}^{m}\Big[\mathcal{N}(r_{ij}|g(U_i^T V_j + P_i^T \mathbf{E} f_j),\sigma_R^2)\Big]^{I_{ij}^R} \\ &\times \prod_{i=1}^{n}\mathcal{N}(U_i|0,\sigma_U^2 I) \times \prod_{j=1}^{m}\mathcal{N}(V_j|0,\sigma_f^2 I) \\ &\times \prod_{i=1}^{n}\mathcal{N}(P_i|0,\sigma_P^2 I) \times \prod_{p=1}^{D}\prod_{q=1}^{C}\mathcal{N}(\mathbf{E}_{pq}|0,\sigma_{\mathbf{E}}^2) \end{aligned} \tag{5}$$

4.2 Modeling Also-viewed Relationships

With regard to the *product-affinity matrix* $\mathcal{S}$, we define the conditional distribution over the observed *product-affinity matrix* as:

$$P(\mathcal{S}|V,Z,\sigma_S^2,\sigma_Z^2) = \prod_{j=1}^{m}\prod_{k=1}^{m}\Big[\mathcal{N}(s_{jk}|g(V_j^T Z_k),\sigma_S^2,\sigma_Z^2)\Big]^{I_{jk}^S} \tag{6}$$

where $Z \in \mathbb{R}^{K\times m}$ is the *"also-viewed"* product embedding matrix, with column vector Z_k representing *"also-viewed"* product-specific embedding vector for product k. We model *"also-viewed"* relationship between product j and k by $V_j^T Z_k$ rather than $V_j^T V_k$ in order to reflect the asymmetric nature of the *product-affinity network*. I_{jk}^S is the indicator function that is equal to 1 if product k belongs to one of the *"also-viewed"* products of product j, and 0 otherwise. Again, we place zero-mean spherical Gaussian priors on the hidden variables V and Z as:

$$P(V|\sigma_V^2) = \prod_{j=1}^{m}N(V_j|0,\sigma_V^2 I),\ P(Z|\sigma_Z^2) = \prod_{k=1}^{m}N(Z_k|0,\sigma_Z^2 I) \tag{7}$$

Hence, given Eqs. 6 and 7, we can compute the log-posterior distribution over the hidden variables as:

$$\begin{aligned} &P(V,Z|S,\sigma_V^2,\sigma_Z^2,\sigma_S^2) \propto P(S|V,Z,\sigma_S^2)P(V|\sigma_V^2)P(Z|\sigma_Z^2) \\ &= \prod_{j=1}^{m}\prod_{k=1}^{m}\Big[N(s_{jk}|g(V_j^T Z_k),\sigma_S^2)\Big]^{I_{jk}^S} \\ &\times \prod_{j=1}^{m}N(V_j|0,\sigma_V^2 I)\prod_{k=1}^{m}N(Z_k|0,\sigma_Z^2 I) \end{aligned} \tag{8}$$

4.3 Unified Model

Thus far, we have shown how to independently model the ratings and *"also-viewed"* relationships among products. In this section, we propose a unified model where user ratings with associated product images are combined with *"also-viewed"* product information, which eventually yields high-quality product embeddings. Given an observed product j rated by user i, Figure 2 illustrates how we fuse both the rating model introduced in Section 4.1 and the *"also-viewed"* relationship model introduced in Section 4.2 into a matrix co-factorization framework by sharing the product embedding matrix V.

Based on Figure 2, the log-posterior distribution of VMCF is given by:

$$\begin{aligned} &P(U,V,P,Z,\mathbf{E}|R,S,\sigma_U^2,\sigma_V^2,\sigma_P^2,\sigma_Z^2,\sigma_{\mathbf{E}}^2,\sigma_R^2,\sigma_S^2) \\ &\propto P(\mathcal{R}|U,V,P,\mathbf{E},\sigma_R^2) \times P(S|V,Z,\sigma_S^2) \\ &P(U|\sigma_U^2)P(V|\sigma_V^2)P(P|\sigma_P^2)P(Z|\sigma_Z^2)P(\mathbf{E}|\sigma_{\mathbf{E}}^2) \\ &= -\frac{1}{2\sigma_R^2}\sum_{i=1}^{n}\sum_{j=1}^{m} I_{ij}^R\Big(r_{ij} - g(U_i^T V_j + P_i^T \mathbf{E} f_j)\Big)^2 \\ &- \frac{1}{2\sigma_S^2}\sum_{j=1}^{m}\sum_{k=1}^{m} I_{jk}^S\Big(s_{jk} - g(V_j^T Z_k)\Big)^2 \\ &- \frac{1}{2\sigma_U^2}\sum_{i=1}^{n} U_i^T U_i - \frac{1}{2\sigma_V^2}\sum_{j=1}^{m} V_i^T V_i - \frac{1}{2\sigma_P^2}\sum_{i=1}^{n} P_i^T P_i \\ &- \frac{1}{2\sigma_Z^2}\sum_{k=1}^{m} Z_k^T Z_k - \frac{1}{2\sigma_{\mathbf{E}}^2}\sum_{p=1}^{D}\sum_{q=1}^{C}\mathbf{E}_{pq}^2 \\ &- \frac{1}{2}\left(\left(\sum_{i=1}^{n}\sum_{j=1}^{m} I_{ij}^R\right)\ln\sigma_R^2 + \left(\sum_{j=1}^{m}\sum_{k=1}^{m} I_{jk}^S\right)\ln\sigma_S^2\right) \\ &- \frac{1}{2}\left(nK\ln\sigma_U^2 + mK(\ln\sigma_V^2 + \ln\sigma_Z^2) + nD\ln\sigma_P^2 + DC\ln\sigma_{\mathbf{E}}^2\right) + \mathcal{C} \end{aligned} \tag{9}$$

where $\mathcal{C}$ is a constant that is independent from the variables to be learned. Maximizing the log-posterior over the hidden variables with fixed hyper-parameters (i.e., the observation noise variances and prior variances) is equivalent to minimizing the following objective function:

$$\begin{aligned} &L(R,S,U,V,P,Z,\mathbf{E}) = \\ &\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{m} I_{ij}^R\Big(r_{ij} - g(U_i^T V_j + P_i^T \mathbf{E} f_j)\Big)^2 \\ &+ \frac{\lambda_S}{2}\sum_{j=1}^{m}\sum_{k=1}^{m} I_{jk}^S\Big(s_{jk} - g(V_j^T Z_k)\Big)^2 \\ &+ \frac{\lambda_U}{2}||U||_F^2 + \frac{\lambda_V}{2}||V||_F^2 + \frac{\lambda_P}{2}||P||_F^2 + \frac{\lambda_Z}{2}||Z||_F^2 + \frac{\lambda_{\mathbf{E}}}{2}||\mathbf{E}||_F^2 \end{aligned} \tag{10}$$

where $\lambda_S = \sigma_R^2/\sigma_S^2$, $\lambda_U = \sigma_R^2/\sigma_U^2$, $\lambda_V = \sigma_R^2/\sigma_V^2$, $\lambda_P = \sigma_R^2/\sigma_P^2$, $\lambda_Z = \sigma_R^2/\sigma_Z^2$, $\lambda_{\mathbf{E}} = \sigma_R^2/\sigma_{\mathbf{E}}^2$ and $||\cdot||_F^2$ denotes the

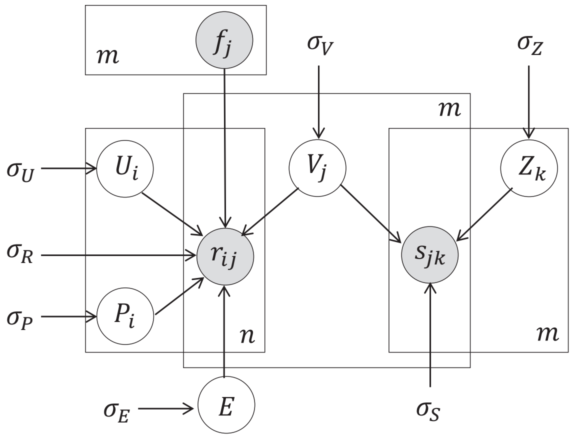

Figure 2: Graphical Model for Visual Matrix Co-Factorization (VMCF), where shaded nodes denote observed variables and the rest, hidden variables.

Frobenius norm. Note that λ_S is a balancing parameter that regulates the importance of *"also-viewed"* products in the unified model. Having formulated a non-convex objective function as Eq. 10, we compute the gradient of each embedding variable, i.e., U_i, V_j, P_i, Z_k, $\mathbf{E}$, and learn them by gradient descent to obtain a local minimum solution. Refer to Appendix A for the time complexity of VMCF and Appendix B for the detailed equations for the gradients.

Reduced Model. Note that when images of products are not available, the final objective function is reduced to:

$$L(R,S,U,V,Z) = \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{m} I_{ij}^{R}\Big(r_{ij} - g(U_i^T V_j)\Big)^2 \\ + \frac{\lambda_S}{2}\sum_{j=1}^{m}\sum_{k=1}^{m} I_{jk}^{S}\Big(s_{jk} - g(V_j^T Z_k)\Big)^2 \\ + \frac{\lambda_U}{2}||U||_F^2 + \frac{\lambda_V}{2}||V||_F^2 + \frac{\lambda_Z}{2}||Z||_F^2 \tag{11}$$

As previously mentioned, in domains where appearance of a product is not significant for modeling user ratings, such product aspects as functionality and specifications are more influential to user ratings than appearance. Thus, the reduced model as in Eq. 11 can be still beneficial, especially for the product domains where the appearance is not important. We dub this model Matrix Co-Factorization (MCF), and demonstrate the benefit of the reduced model in product domains where appearance is not important in Section 5.4.

5. EXPERIMENTS

In this section, we conduct experiments to verify the superiority of our methods by comparing their performance with several state-of-the-art methods on multiple real-world datasets. The experiments are designed to verify the following questions:

Q.1 How do MCF and VMCF perform compared with other competitors in both the visually-aware (Boys' and Girls' Clothing) and the visually non-aware product domains (Automotive, Pet Supplies and Office Products)?

Table 2: Data Statistics. #Relations implies the number of edges in the *product-affinity network*

Dataset	#Users	#Prod.	#Ratings	#Relations
Boys' Clothing	4,496	6,391	15,997	31,370
Girls' Clothing	5,941	9,549	22,524	51,990
Automotive	84,418	126,934	406,852	2,162,853
Pet Supplies	85,115	49,048	427,543	1,066,131
Office Prod.	50,570	40,181	240,146	672,586

Table 3: Properties of methods being compared.

Baselines	Personalized?	Visually-Aware?	Incorporate *"also-viewed"*?
ItemCF	X	X	X
PMF	O	X	X
VMF	O	O	X
MCF	O	X	O
VMCF	O	O	O

Q.2 By leveraging *"also-viewed"* products, do we indeed obtain high-quality product embedding?

Q.3 How does model parameter λ_S and the number of embedding dimensions K affect the user rating prediction accuracy?

5.1 Datasets

We use five public datasets[2] extracted from *Amazon.com* by McAuley *et al.* [18]. The datasets include user ratings data and product metadata, which includes URLs for product image, price, a list of also bought product, a list of also viewed products, and etc. Among these metadata, we use the lists of also viewed products to construct our *product-affinity network*. Aiming at demonstrating the benefit of our model in general product domains, we not only use two datasets (*Boys' Clothing and Girls' Clothing*) from domains where the appearance of a product is significant, but also three datasets (*Automotive, Pet Supplies and Office Products*) from domains where other product aspects, such as product functionality and specifications, play a more significant role than the appearance. We preprocess all datasets so that each user rated at least three products. Moreover, every product in the *product-affinity network* also exists in the user ratings data. Table 2 shows the detailed statistics of the datasets.

5.2 Comparison Methods

- **ItemCF:** A traditional recommendation method based on the similarity of products [24], where Pearson's correlation coefficient is used as similarity measure.

- **PMF:** Matrix factorization-based recommendation method that considers only user ratings information [23].

[2] http://jmcauley.ucsd.edu/data/amazon/

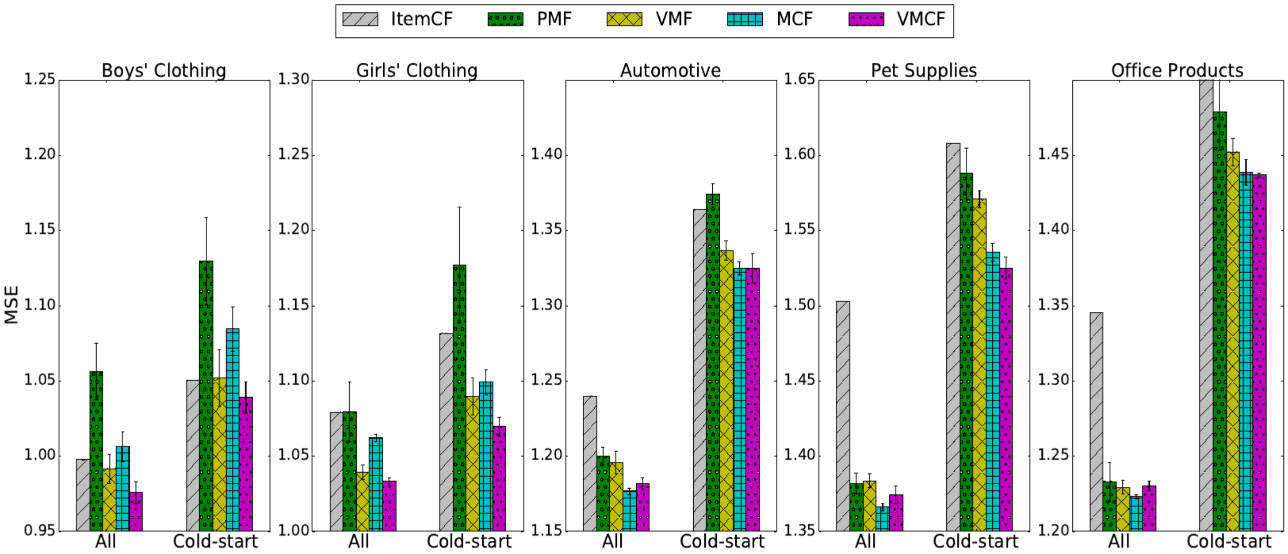

Figure 3: Performance comparison in terms of MSE. Lower MSE implies better performance.

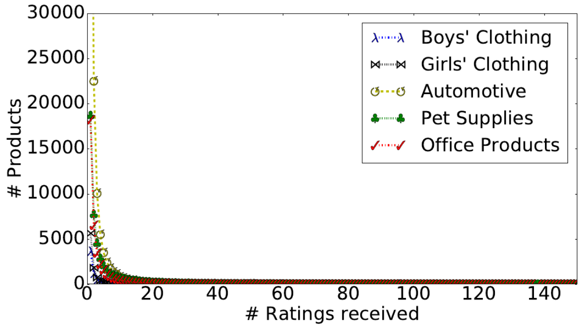

Figure 4: Skewness of the number of rated products in each dataset. Most products received very few ratings.

- **VMF:** A visually-aware matrix factorization-based method that incorporates product images but not *"also-viewed"* product information. We use the rating prediction model introduced in [9], from which bias terms are excluded to clearly investigate the benefit of incorporating product images themselves. The rating assigned by user i to product j is modeled as $\hat{r}_{ij} = U_i^T V_j + P_i^T \mathbf{E} f_j$, and the objective function to minimize is formulated as:

$$L = \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{m} I_{ij}^{R} (r_{ij} - g(\hat{r}_{ij}))^2 + \Omega(U, V, P, \mathbf{E})$$

where $\Omega(\cdot)$ is the regularization to avoid model overfitting.

In order to provide a clear understanding of baseline methods, we provide a summary of their properties in Table 3. Furthermore, these baselines are chosen for the following reasons:

1. **PMF vs. VMF**

   a) To verify the benefit of incorporating product images in both visually-aware and visually non-aware product domain, and b) To verify that appearance is more significant in clothing domain than other domains.

2. **VMF vs. MCF**

   To verify that other aspects besides appearance are more significant in visually non-aware product domain than in visually-aware product domain.

3. **VMF, MCF vs. VMCF**

   To demonstrate the benefit of jointly modeling product images and *"also-viewed"* product information in both visually-aware and visually non-aware product domain.

5.3 Experimental Settings

Evaluation Metric. We employ the Mean Squared Error (MSE), a metric that has been commonly used for evaluating the performance of user rating prediction on the *Amazon* dataset [2, 14, 16]. The MSE is defined as:

$$MSE = \frac{\sum_{i,j} (r_{ij} - \hat{r}_{ij})^2}{N} \quad (12)$$

where r_{ij} denotes the rating that user i assigned to product j, $\hat{r}_{ij}$ denotes the corresponding predicted rating, and N denotes the number of ratings in the test dataset. The MSE is an appropriate metric for our experiments because the objective functions of the methods being compared are originally designed to minimize the MSE.

Evaluation Protocol. We randomly sample 80% of the user ratings datasets for training, and the remaining 20% is used for testing. Random sampling is independently conducted five times, and we evaluate the baselines and our methods on each dataset. Finally, we report here the mean and standard deviation (error bar) of the MSE on each test dataset.

Cold-start Evaluation. In addition to the *"All"* setting where we evaluate on all the ratings in the test dataset, we also evaluate our methods on the *"Cold-start"* setting where ratings in the test dataset are sampled such that each product (cold-product) has fewer than four ratings in the training dataset. Note that as shown in Figure 4, most products received very few ratings. Precisely, cold-products constitute approximately 85% in Boys' Clothing, Girls' Clothing and Automotive, 70% in Pet Supplies and Office Products. These statistics indicate that most products are cold-products in the real-world online shopping environment, and thus *evaluations on cold-products are in fact more crucial than evaluations on every product.*

Parameters. For ItemCF, we set the number of neighbors to 20. For all other matrix factorization-based methods, in order to find the best parameters for each dataset, we perform grid search with $\lambda_U, \lambda_V, \lambda_P \in \{0.01, 0.1, 0.5, 1.0\}$, $\lambda_Z \in \{0.1, 0.01\}$, $\lambda_\mathbf{E} \in \{0.1, 1, 3, 7, 9, 10, 11\}$ and $\lambda_S \in \{0.01, 0.1, 0.5, 1.0\}$ while K and D are fixed to 5. As a result, while $\lambda_U = \lambda_V = \lambda_P = 0.1, \lambda_Z = 0.01$ exhibit the best performance for all datasets, $\lambda_\mathbf{E} = 7$ and $\lambda_\mathbf{S} = 0.2$ yield the best results for *Boys' Clothing*, $\lambda_\mathbf{E} = 7$ and $\lambda_\mathbf{S} = 0.1$ for *Girls' Clothing*, $\lambda_\mathbf{E} = 11$ and $\lambda_\mathbf{S} = 0.4$ for *Automotive*; $\lambda_\mathbf{E} = 11$ and $\lambda_\mathbf{S} = 1.0$ for *Pet Supplies*, and $\lambda_\mathbf{E} = 10$ and $\lambda_\mathbf{S} = 0.1$ for *Office Products.*

5.4 Performance Analysis (Q.1)

Figure 3 summarizes the evaluation results on each dataset in terms of MSE where performance is evaluated under two settings, i.e., *"All"* and *"Cold-start"*. Note that visually-aware product domain denotes Boys' Clothing and Girls' Clothing, and visually non-aware product domain denotes Automotive, Office Products and Pet Supplies.

1) Benefit of visual features in both domains (PMF vs. VMF).
We observe that in both the visually-aware and the visually non-aware product domain, *incorporating visual features helps to improve user rating prediction accuracy*, while the improvement is more significant in visually-aware product domain. This agrees with our expectations whereby a) appearance plays a more significant role in modeling user ratings in visually-aware product domain, and b) although other product aspects are more influential in visually non-aware product domain, visual features are indeed still helpful in modeling user ratings. Note that the *performance gain of VMF compared with PMF becomes more significant under the "Cold-start" setting*, which indicates that visual features become more valuable when a product only has a few ratings.

2) Varying significance of product aspects in different domains (VMF vs. MCF).
We observe that *in the visually-aware product domain, the user rating prediction accuracy of VMF outperforms MCF, whereas MCF outperforms VMF in the visually non-aware product domain.* This agrees with **Example 1** in Section 1, in that appearance plays the most significant role in visually-aware product domain, whereas other product aspects beyond appearance play a more significant role in visually non-aware product domain.

3) Benefit of jointly modeling product images and *"also-viewed"* product information (VMF, MCF vs. VMCF).
We observe that *in the visually-aware product domain, VMCF outperforms both VMF and MCF.* This indicates that in the visually-aware product domain, other aspects beyond appearance of products are also taken into account by modeling the *"also-viewed"* relationship, which in turn yields further improvements in user rating prediction accuracy. Such a benefit becomes clearer in the *"Cold-start"* setting.

Meanwhile, *under the "All" setting in the visually non-aware product domain, VMCF fails to outperform MCF.* This implies that when a sufficient number of ratings is assigned to each product, visual features in the visually non-aware product domain are in fact considered as *noise*, and thus degrade user rating prediction accuracy. However, it is worth noting that *under the "Cold-start" setting, the performance of VMCF is comparable to that of MCF or even better in Pet Supplies and Office Products.* This indicates that visual features remain helpful even in the visually non-aware product domain, when we are provided with only a few rated products.

5.5 Quality of Product Embedding (Q.2)

Besides the performance evaluations in terms of the accuracy of user rating prediction, we additionally evaluate the quality of product embedding generated by our proposed method in comparison with VMF. To do so, we perform classification on both product embedding $V \in \mathcal{R}^{K\times m}$ and product visual embedding $Q \in \mathcal{R}^{D\times m}$ whose goal is to classify which category a product belongs to. In order to demonstrate that the classification results are reliable and robust, we select four classification algorithms, i.e., Logistic regression (LR), Support vector machine (SVM), Random forest (RF) and Gradient boosting (GB). The input products for classification algorithms belong to the top-10 most frequently appeared categories in each dataset, and each product is labeled by its corresponding category. Refer to Appendix C for the list of these categories in each dataset. We perform five-fold cross-validation and report the mean and standard deviation (error bar).

Figure 5 shows the results of classification. We observe that in the visually non-aware product domain, classification accuracy on product embedding V and the product visual embedding Q generated by VMCF outperform those generated by VMF. This indicates that *VMCF generates more high-quality embeddings than VMF, which eventually results in higher accuracy in user rating prediction* as shown in Figure 3. Meanwhile, in the visually-aware product domain, while VMCF outperforms VMF in the user rating prediction task, the classification accuracy of VMCF on V is higher than that of VMF, but its the accuracy on Q is lower. We conjecture that using only product images is helpful for the product classification task because visual features are generated by a pre-trained CNN originally designed for image classification task. In contrast, for our user rating prediction task, *product embedding V, which models the "also-viewed" product information where various product aspects are reflected, works in synergy with Q to improve user rating prediction accuracy.*

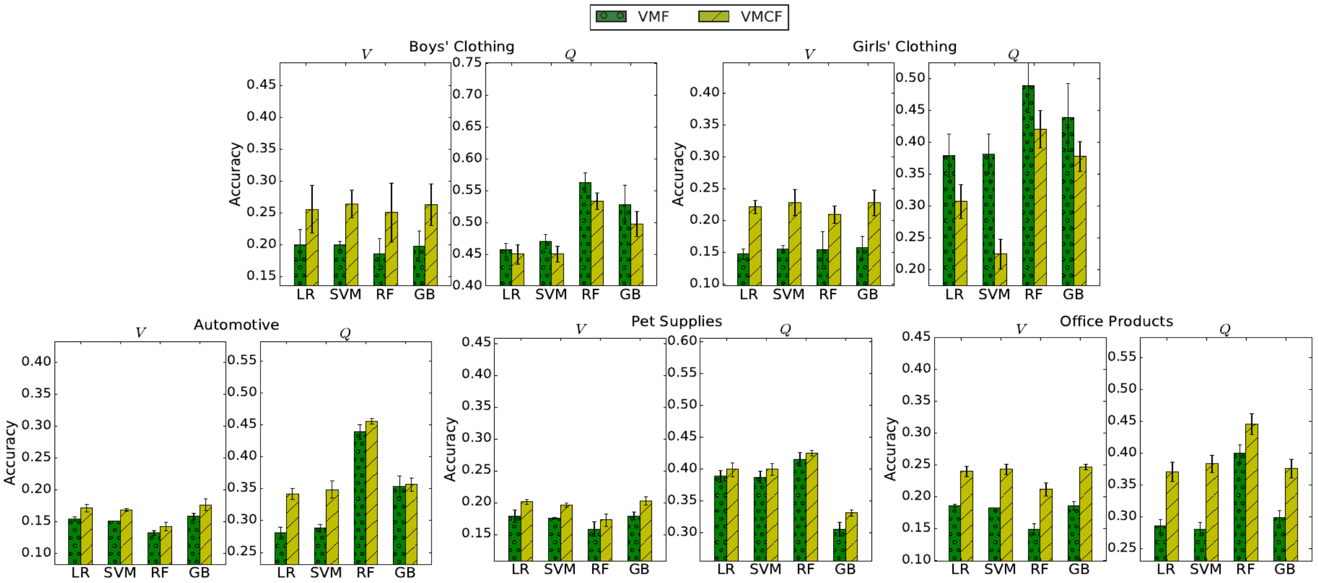

Figure 5: Classification results on product embedding V and product visual embedding Q.

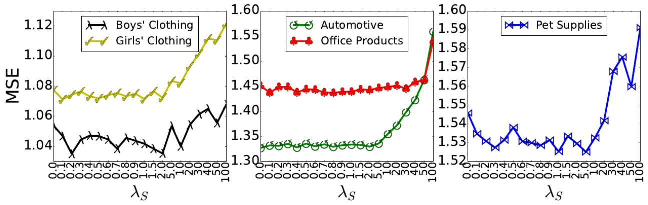

Figure 6: Impact of Parameter λ_S on VMCF.

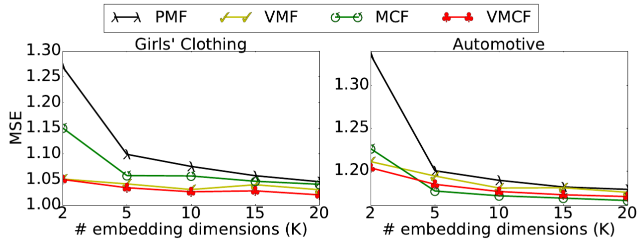

Figure 7: Results of various number of embedding dimensions K.

5.6 Sensitivity Analysis (Q.3)

Influence of balancing parameter λ_S

λ_S is a hyper-parameter that regulates the importance of *"also-viewed"* product information in our proposed method VMCF. When $\lambda_S = 0$, we ignore *"also-viewed"* product information, whereas when $\lambda_S = \infty$, we only exploit *"also-viewed"* product information. Figure 6 shows the evaluation results of VMCF performed on each dataset under the *"Cold-start"* setting. We observe that incorporating *"also-viewed"* product information indeed improves user rating prediction performance, and the optimal value of λ_S is different in each dataset, which is mostly a value in the range [0.0, 1.0].

Influence of num. dimensions K of product embedding

Figure 7 shows the evaluation results on the test set of Girls' Clothing and Automotive over different numbers of embedding dimensions K under the *"All"* setting, given a fixed $D = 5$. We observe that for all methods (PMF, VMF, MCF and VMCF) the value of MSE decreases as the number of latent dimensions increases. However, since the total complexity of VMCF is linear in the number of embedding dimensions as demonstrated in Appendix A, a proper value of K should be found such that complexity is practically acceptable within computational limitations. Evaluations on other datasets yield similar results, and are thus omitted here for brevity.

6. CONCLUSION & FUTURE WORK

Every product domain has dominant aspects that are more influential to user ratings than others. In clothing domain, the appearance of products plays the most significant role, whereas in other domains such as Automotive, Pet Supplies and Office Products, other aspects such as product functionality and specifications are more influential. In this paper, we propose a matrix co-factorization framework that jointly factorizes user ratings data and *"also-viewed"* product information that reflects various product aspects that are varyingly influential to user ratings in different product domains. We empirically show that this information is helpful for user rating prediction by generating more high-quality product embedding, especially under the *"Cold-start"* setting. Our method is useful for online retailers such as *Amazon* and *eBay*[3], where product images are provided to customers and their browsing histories are collected. Although we only leverage *"also-viewed"* product information in this paper, other relationships among products are also prevalent, such as *"also-bought"*, *"frequently-bought-together"* or *"bought-after-buying"*. In the future, we plan to investigate whether these relationships are helpful in generating even more high-quality product embedding, and if so, how to properly integrate them into our system to model user ratings.

[3] http://www.ebay.com/

7. ACKNOWLEDGMENTS

This work was supported by 1) the Industrial Core Technology Development Program (10049079 , Development of Mining core technology exploiting personal big data) funded by the Ministry of Trade, Industry and Energy (MOTIE, Korea), 2) the Next-Generation Information Computing Development Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education, Science and Technology (No. 2012M3C4A7033344) and 3) ICT R&D program of MSIP/IITP [14-824-09-014, Basic Software Research in Human-level Lifelong Machine Learning (Machine Learning Center)]

APPENDIX

A. COMPLEXITY ANALYSIS

The overall complexity of VMCF is composed of the calculation of both Eqs. 10 and 13. Considering the sparseness of $\mathcal{R}$ and $\mathcal{S}$, the computation Eq. 10 has complexity $O(\rho(K + D) + \mu K)$, where ρ denotes the average number of observed ratings in $\mathcal{R}$, and μ denotes the average number of observed elements in $\mathcal{S}$. Next, for the gradients in Eq. 13, computing $\frac{\partial L}{\partial U_i}$, $\frac{\partial L}{\partial V_j}$, $\frac{\partial L}{\partial P_i}$, $\frac{\partial L}{\partial Z_k}$ and $\frac{\partial L}{\partial \mathbf{E}}$ incur complexity $O(\rho K)$, $O(\rho K + \mu K)$, $O(\rho D\bar{C}) = O(\rho D)$, $O(\mu K)$ and $O(\rho D\bar{C}) = O(\rho D)$, respectively, where $\bar{C}$ is the average number of non-zero elements in a CNN feature (f_j). Note that $\bar{C}$ is a small value, since the CNN feature is very sparse. Thus, we obtain a total complexity of $O(\rho(K + D) + \mu K)$. This analysis indicates that the complexity of VMCF is linear in the number of embedding dimensions.

B. GRADIENTS OF EQUATION 10

$$\begin{aligned}
\frac{\partial L}{\partial U_i} &= \sum_{j=1}^{m} I_{ij}^{R}(g(\hat{r}_{ij}) - r_{ij})g'(\hat{r}_{ij})V_j + \lambda_U U_i \\
\frac{\partial L}{\partial V_j} &= \sum_{i=1}^{n} I_{ij}^{R}(g(\hat{r}_{ij}) - r_{ij})g'(\hat{r}_{ij})U_i \\
&\quad + \lambda_S \sum_{k=1}^{m} I_{jk}^{S}(g(\hat{s}_{jk}) - s_{jk})g'(\hat{s}_{jk})Z_k + \lambda_V V_j \\
\frac{\partial L}{\partial P_i} &= \sum_{j=1}^{m} I_{ij}^{R}(g(\hat{r}_{ij}) - r_{ij})g'(\hat{r}_{ij})\mathbf{E}f_j + \lambda_P P_i \\
\frac{\partial L}{\partial Z_k} &= \lambda_S \sum_{j=1}^{m} I_{jk}^{S}(g(\hat{s}_{jk}) - s_{jk})g'(\hat{s}_{jk})V_j + \lambda_Z Z_k \\
\frac{\partial L}{\partial \mathbf{E}} &= \sum_{i=1}^{n}\sum_{j=1}^{m} I_{ij}^{R}(g(\hat{r}_{ij}) - r_{ij})g'(\hat{r}_{ij})P_i f_j^T + \lambda_{\mathbf{E}}\mathbf{E}
\end{aligned} \tag{13}$$

where $g'(x) = exp(x)/(1 + exp(x))^2$ denotes the derivative of the logistic function.

C. TOP-10 MOST FREQUENT CATEGORIES

- *Boys' Clothing:* Athletic, Boots, Sneakers, Tops & Tees, Costumes & Accessories, Sandals, Kids & Baby, Pants, Shirts, Hoodies
- *Girls' Clothing:* Boots, Athletic, Sneakers, Costumes & Accessories, T-Shirts, Jewelry, Kids & Baby, Sandals, Special Occasion, Playwear
- *Automotive:* 'Shocks, Struts & Suspension', 'Paint, Body & Trim', Filters, Brake System, Protective Gear, Bulbs, Decals & Bumper Stickers, Floor Mats & Cargo Liners, Towing Products & Winches, Car Care
- *Pet Supplies:* 'Collars, Harnesses & Leashes', Food, Health Supplies, Toys, Treats, Apparel & Accessories, Pumps & Filters, Beds, Carriers & Travel Products, Shampoos & Conditioners
- *Office Products:* Pens & Refills, Paper, Inkjet Printer Ink, 'Labels, Indexes & Stamps', Laser Printer Toner, Office Furniture & Lighting, 'Envelopes, Mailers & Shipping Supplies', Notebooks & Writing Pads, Telephones & Accessories, 'Forms, Recordkeeping & Money Handling'

D. REFERENCES

[1] D. Agarwal and B.-C. Chen. flda: matrix factorization through latent dirichlet allocation. In *Proceedings of the third ACM international conference on Web search and data mining*, pages 91–100. ACM, 2010.

[2] Y. Bao, H. Fang, and J. Zhang. Topicmf: simultaneously exploiting ratings and reviews for recommendation. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, pages 2–8. AAAI Press, 2014.

[3] O. Barkan and N. Koenigstein. Item2vec: Neural item embedding for collaborative filtering. *arXiv preprint arXiv:1603.04259*, 2016.

[4] J. Bennett and S. Lanning. The netflix prize. In *Proceedings of KDD cup and workshop*, volume 2007, page 35, 2007.

[5] C. Chen, X. Zheng, Y. Wang, F. Hong, Z. Lin, et al. Context-aware collaborative topic regression with social matrix factorization for recommender systems. In *AAAI*, volume 14, pages 9–15, 2014.

[6] M. D. Ekstrand, J. T. Riedl, and J. A. Konstan. Collaborative filtering recommender systems. *Foundations and Trends in Human-Computer Interaction*, 4(2):81–173, 2011.

[7] G. Guo, J. Zhang, and N. Yorke-Smith. Trustsvd: collaborative filtering with both the explicit and implicit influence of user trust and of item ratings. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*. AAAI Press, 2015.

[8] R. He, C. Lin, J. Wang, and J. McAuley. Sherlock: Sparse hierarchical embeddings for visually-aware one-class collaborative filtering. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*, pages 3740–3746, 2016.

[9] R. He and J. McAuley. VBPR: visual bayesian personalized ranking from implicit feedback. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA.*, pages 144–150, 2016.

[10] D. Kim, C. Park, J. Oh, S. Lee, and H. Yu. Convolutional matrix factorization for document context-aware recommendation. In *Proceedings of the 10th ACM Conference on Recommender Systems*, RecSys '16, pages 233–240, New York, NY, USA, 2016. ACM.

[11] Y. Koren. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 426–434. ACM, 2008.

[12] Y. Koren. Collaborative filtering with temporal dynamics. *Commun. ACM*, 53(4):89–97, Apr. 2010.

[13] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[14] G. Ling, M. R. Lyu, and I. King. Ratings meet reviews, a combined approach to recommend. In *Proceedings of the 8th ACM Conference on Recommender systems*, pages 105–112. ACM, 2014.

[15] H. Ma, D. Zhou, C. Liu, M. R. Lyu, and I. King. Recommender systems with social regularization. In *Proceedings of the Fourth ACM International Conference on Web Search and Data Mining*, WSDM '11, pages 287–296, New York, NY, USA, 2011. ACM.

[16] J. McAuley and J. Leskovec. Hidden factors and hidden topics: understanding rating dimensions with review text. In *Proceedings of the 7th ACM conference on Recommender systems*, pages 165–172. ACM, 2013.

[17] J. McAuley, R. Pandey, and J. Leskovec. Inferring networks of substitutable and complementary products. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2015.

[18] J. McAuley, C. Targett, Q. Shi, and A. van den Hengel. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 43–52. ACM, 2015.

[19] C. Park, D. Kim, J. Oh, and H. Yu. Improving top-k recommendation with truster and trustee relationship in user trust network. *Information Sciences*, 374:100 – 114, 2016.

[20] C. Park, D. Kim, J. Oh, and H. Yu. Trecso: Enhancing top-k recommendation with social information. In *Proceedings of the 25th International Conference Companion on World Wide Web*, pages 89–90, 2016.

[21] S. Purushotham, Y. Liu, and C.-c. J. Kuo. Collaborative topic regression with social matrix factorization for recommendation systems. In *Proceedings of the 29th International Conference on Machine Learning (ICML-12)*, pages 759–766, 2012.

[22] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.

[23] R. Salakhutdinov and A. Mnih. Probabilistic matrix factorization. In *Advances in Neural Information Processing Systems*, volume 20, 2008.

[24] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web*, pages 285–295. ACM, 2001.

[25] C. Wang and D. M. Blei. Collaborative topic modeling for recommending scientific articles. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 448–456. ACM, 2011.

[26] H. Wang, N. Wang, and D.-Y. Yeung. Collaborative deep learning for recommender systems. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1235–1244. ACM, 2015.

[27] L. Xiong, X. Chen, T.-K. HUang, J. Schneider, and J. G. Carbonell. Temporal collaborative filtering with bayesian probabilistic tensor factorization. In *Proceedings of the 2010 SIAM International Conference on Data Mining*, pages 211–222, 2010.