The VLDB Journal (2021) 30:495–513
https://doi.org/10.1007/s00778-021-00651-y

REGULAR PAPER


# Visually aware recommendation with aesthetic features

Wenhui Yu[1] · Xiangnan He[2] · Jian Pei[3] · Xu Chen[4] · Li Xiong[5] · Jinfei Liu[6] · Zheng Qin[7]

Received: 9 February 2020 / Revised: 2 December 2020 / Accepted: 9 January 2021 / Published online: 27 February 2021


**Abstract**
Visual information plays a critical role in human decision-making process. Recent developments on visually aware recommender systems have taken the product image into account. We argue that the aesthetic factor is very important in modeling and predicting users' preferences, especially for some fashion-related domains like clothing and jewelry. This work is an extension of our previous paper (Yu et al., in: WWW, pp 649–658, 2018), where we addressed the need of modeling aesthetic information in visually aware recommender systems. Technically speaking, we make three key contributions in leveraging deep aesthetic features. In Yu et al. (in: WWW, pp 649–658, 2018), (1) we introduced the *aesthetic features* extracted from product images by a deep aesthetic network to describe the aesthetics of products. We incorporated these features into recommender system to model users' preferences in the aesthetic aspect. (2) Since in clothing recommendation, time is very important for users to make decision, we designed a new tensor decomposition model for implicit feedback data. The aesthetic features were then injected to the basic tensor model to capture the temporal dynamics of aesthetic preferences. In this extended version, we try to explore aesthetic features in negative sampling to get further benefit in recommendation tasks. In implicit feedback data, we only have positive samples. Negative sampling is performed to get negative samples. In conventional sampling strategy, uninteracted items are selected as negative samples randomly. However, we may sample potential samples (preferred but unseen items) as negative ones by mistake. To address this gap, (3) we use the aesthetic features to optimize the sampling strategy. We enrich the pairwise training samples by considering the similarity among items in the aesthetic space (and also in the semantic space and graphs). The key idea is that a user may likely have similar perception on similar items. We perform extensive experiments on several real-world datasets and demonstrate the usefulness of aesthetic features and the effectiveness of our proposed methods.

**Keywords** Item recommendation · Side information · Aesthetic features · Tensor factorization · Pairwise learning to rank

## 1 Introduction

Recommender systems have been widely used in online services to predict users' preferences based on their interaction histories [16]. Recently, visual information has been intensively explored to enhance the performance of recommender models [7,9,14,46]. In many domains of interest, the images of items play an important role in user decision-making process. For example, when purchasing clothing, users will

✉ Xu Chen
xu.chen@ruc.edu.cn

Wenhui Yu
jianlin.ywh@alibaba-inc.com

Xiangnan He
xiangnanhe@gmail.com

Jian Pei
jpei@cs.sfu.ca

Li Xiong
lxiong@emory.edu

Jinfei Liu
jinfeiliu@zju.edu.cn

Zheng Qin
qingzh@mail.tsinghua.edu.cn

1 Alibaba Group, Beijing, China

2 University of Science and Technology of China, Hefei, China

3 Simon Fraser University, Surrey, Canada

4 Beijing Key Laboratory of Big Data Management and Analysis Methods, Gaoling School of Artificial Intelligence, Renmin University of China, Beijing, China

5 Emory University, Atlanta, USA

6 Zhejiang University, Hangzhou, China

7 Tsinghua University, Beijing, China

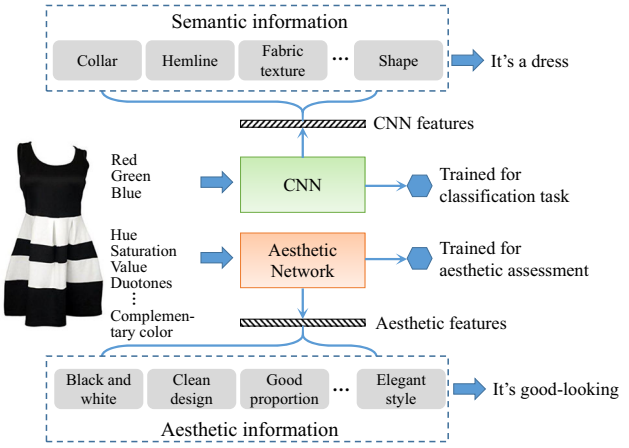



**Fig. 1** Comparison of CNN features and aesthetic features. The CNN is inputted with the RGB components of an image and trained for the classification task, while the aesthetic network is inputted with raw aesthetic features and trained for the aesthetic assessment task

scrutinize product images for the intuitive representation of the clothing like shape, design, color schemes, decorative pattern, texture, and so on. To leverage these kinds of information, existing efforts have extracted various visual features from item images and injected them into recommender models, like SIFT features, CNN features, color histograms, etc. For example, Zhao et al. [46], Chen et al. [7] utilized low-level SIFT features and color histograms, and He and McAuley [14], McAuley et al. [29], Chen et al. [9] utilized high-level CNN features extracted by a deep convolutional neural network.

We argue that the aesthetic information is crucial in predicting user preferences on products in many domains, such as clothing, furniture, ornaments, and electronics. Taking the product shown in Fig. 1 as an example, besides the semantic information, a user will also notice that the dress is with colors black and white, simple yet elegant design, and delightful proportion. She may have the intention to purchase it if she is satisfied with these aesthetic factors. In fact, for many users, especially young females, the aesthetic factor could be the primary factor when purchasing clothes. Unfortunately, conventional visual features do not encode the aesthetic information by nature. Zhao et al. [46] used color histograms to portray users' intuitive perception about an image, but the solution leaves much space to improve, since it does not make good use of many valuable information, such as aesthetic information shown in Fig. 1. To address this issue, we proposed a more comprehensive and high-level aesthetic representation for items in our previous paper [43].

This paper is the extension of [43], where we extracted aesthetic-related features with a dedicated neural network called **B**rain-inspired **D**eep **N**etwork (**BDN**) [39]. We input raw features that are indicative of human aesthetic feelings, such as hue, saturation, duotones, and complementary colors, and train BDN for the image aesthetic assessment task. We use the backbone to extract high-level aesthetic features. Intuitively, BDN is trained to mine information that is important to the aesthetic assessment task, thus these features encode aesthetic factors such as colors, structure, proportion, and styles (see Fig. 1 as an example). In this paper, to make a thorough use of aesthetic features, we additionally utilize them for negative sampling. In our method, the aesthetic features are used for both modeling and learning: we define the user preference model to be aware of aesthetic features, and then use them to improve the sampling quality when learning the model.

We first introduce the research effort in [43]. Compared with other products, clothing shows obvious temporal characteristics, since in clothing recommendation, if an item can be purchased depends on not only if the user likes the it, but also if it fits the current time. To design the basic model, we consider these two vital factors. Also, users' aesthetic preferences are impacted by these two factors: (1) It is obvious that aesthetic preferences show a significant diversity among different people. For instance, when purchasing clothing, children prefer colorful and lovely products while adults prefer those can make them look mature and elegant (for empirical evidence, see Fig. 7); women may prefer exquisite decorations while men like concise designs (see Fig. 8). (2) The aesthetic tastes of users also change over time, either in short term, or in long term. For example, the aesthetic tastes vary in different seasons periodically—in spring or summer, people may prefer clothes with light color and fine texture, while in autumn or winter, people tend to buy clothes with dark color, rough texture, and loose style (see Fig. 9). In the long term, the fashion trend changes all the time and the popular colors and design may be different by year (see Fig. 10).

Considering the above-mentioned factors, we exploit tensor factorization as the basic model to capture the diversity of aesthetic preferences among users and over time. There are several ways to decompose a tensor [21,35], however, there are certain drawbacks in the existing models. To tailor it for the clothing recommendation task, we propose a new tensor factorization model trained with coupled matrices to mitigate the *sparsity* problem [2]. We then combine the basic model with the additional visual features (concatenated aesthetic and CNN features) and term the method **V**isually Aware **R**ecommendation with **A**esthetic Features (**VRA**).

Now, we introduce the research effort in this paper. The other technical contribution of the paper lies in the learning part. We not only leverage the features we extracted to model users' aesthetic preference in VRA, but also improve the quality of negative sampling by measuring the similarity of items in the aesthetic space. When optimizing a model on *implicit feedback* data (e.g., purchasing records), pairwise learning has been widely used due to its rationality, which aims to maximize the margin between the predictions of pos-

itive and negative samples [36]. In this paper, we design a pairwise learning to rank method to factorize the tensor and coupled matrices. However, when employing pairwise learning, one critical issue is that not all unobserved feedbacks are necessarily negative samples, since some of them might be just unknown by users, i.e., potential positive samples while mislabeled as negative ones. To address this issue, we construct the neighbor set of each item by finding the similar items in the aesthetic space. The intuition is that the items in the neighbor set of a purchased item (positive sample) are more likely to fit the user's aesthetics thus are more likely to be potential positive samples. We treat these potential positive samples as the third kinds of labels between the positive and negative ones in our **A**esthetic-enhanced **P**airwise **L**earning to **R**ank (**APLR**) algorithm.

Finally, we evaluate the performance of our proposed method by comparing it with several baselines on an *Amazon* dataset and 5 subsets. Extensive experiments show that the recommendation accuracy can be significantly improved by incorporating aesthetic features. To summarize, our main contributions are as follows:

- We propose aesthetic features for items' aesthetic representation, and then leverage these features in the recommendation context. Moreover, we compare the effectiveness with several conventional features to demonstrate the necessity of the aesthetic features.
- We propose a new tensor factorization model to portray the purchase events in three dimensions: users, items, and time. We then inject the aesthetic features into it to model users' aesthetic preference.
- We use the aesthetic features to enhance the optimization strategy for the proposed model. To enrich pairwise training samples, we construct neighbor set for positive items by considering the similarity between items evidenced by visual features and collaborative information. This is the main contribution compared with our previous paper [43].
- We validate the effectiveness of our proposed model by comparing it against several state-of-the-art baselines on 6 real-world datasets. Experiments show that we gain significant improvement by exploring aesthetic features in modeling user preference and negative sampling.

## 2 Related work

Recommender systems have gained more and more attention due to their extensive applications, and created considerable economic benefits. On various online platforms such as E-commerce, video, and news online platforms, recommender systems help users to find their interested items efficiently and improve the user experience significantly. Collaborative filtering (CF) model [13,22,37] boosts the development of recommender systems. Among various CF methods, matrix factorization (MF) [22,36], which encodes user preferences by underlying latent factors, is a basic yet the most effective recommender model. To improve the presentation capability, many variants have been proposed [2,14,16,32,41].

This paper develops aesthetic-aware clothing recommender systems. Specifically, we incorporate the features extracted from the product images by an aesthetic network into a tensor factorization model, and optimize our model with pairwise learning. As such, we review related work on aesthetic networks, image-based recommendation, tensor factorization, and negative sampling strategies.

### 2.1 Aesthetic networks

The aesthetic networks are proposed for image aesthetic assessment. After Datta et al. [10] first proposed the aesthetic assessment problem, many research efforts exploited various handcrafted features to extract the aesthetic information of images [10,26,28]. To portray the subjective and complex aesthetic perception, Lu et al. [25], Wang et al. [39], Ma et al. [27] exploited deep networks to emulate the underlying complex neural mechanisms of human perception, and displayed the ability to describe image content from the primitive-level (low-level) features to the abstract-level (high-level) features. Proposed in [39], **B**rain-inspired **D**eep **N**etwork (**BDN**) model is the state-of-the-art aesthetic deep model. In this paper, we use BDN to extract the aesthetic features of product images, and use these features to enhance the performance of the recommender system.

### 2.2 Image-based recommendations

Recommendation has been widely studied due to its extensive use. The power of recommender systems lies on their ability to model complex preferences that users exhibit toward items based on their past interactions and behavior. To extend their expressive power, various works exploited image data [6,7,9,14,46]. For example, McAuley et al. [29], He and McAuley [14] used CNN features of product images while Zhao et al. [46] recommended movies with color histograms of posters and frames. Sha et al. [38], Jagadeesh et al. [18] recommended clothes by considering the clothing fashion style. Though various visual features are leveraged in recommendation tasks, they are conventional features (such as CNN features and SIFT features) and low-level aesthetic features (such as color histograms). To propose more powerful aesthetic features, Yu et al. [43] extracted high-level features by a BDN pretrained for the aesthetic assessment task, and used these features to model users' aesthetic preference. This paper is the extended version of [43]. To explore aesthetic

features in different aspects, we used them to improve the quality of negative sampling.

### 2.3 Tensor factorization

Time is an important contextual information in recommender systems since the sales of commodities show a distinct time-related succession. In context-aware recommender systems, tensor factorization has been extensively used. For example, Kolda and Bader [21] introduced two main forms of tensor decomposition, the **C**ANDECOMP/ **P**ARAFAC (**CP**) and Tucker decomposition. Karatzoglou et al. [20] first utilized tensor factorization for context-aware collaborative filtering. Rendle and Schmidt-Thieme [35] proposed a **P**airwise **I**nteraction **T**ensor **F**actorization (**PITF**) model to decompose the tensor with a linear complexity. Tensor-based methods suffer from several drawbacks like poor convergence in sparse data [4] and not scalable to large-scale datasets [1]. To address these limitations, Acar et al. [2], Bhargava et al. [3] formulated recommendation models with the **C**oupled **M**atrix and **T**ensor **F**actorization (**CMTF**) framework. All existing tensor decomposition models are designed for explicit feedback data and usually do not perform well in implicit feedback cases. In this paper, we design a novel tensor decomposition model for implicit feedback data and incorporate aesthetic features into it.

### 2.4 Negative sampling

In real-world applications, data of implicit feedback, or *one-class* form is easier to collect so extensively used. Prediction on implicit feedback dataset is a challenging task since we only know positive samples and unobserved samples, but cannot discriminate negative samples and potential positive samples from the unobserved ones [15]. In [36], all unobserved samples are treated equally as negative ones when sampling. To improve the sampling quality, many works proposed enhanced pairwise learning with various extra information [11,32–34,45]. For example, Ding et al. [11], Pan et al. [33] used view information to enrich positive samples. Pan and Chen [32], Liu et al. [23] utilized collaborative information mined from the connections of users and items. Cao et al. [5], Liu et al. [24] proposed listwise ranking methods instead of pairwise ones. Yu and Qin [42] considered the noise in negative samples, and optimized the negative sampling strategy based on noisy label-robust learning. Yu et al. [44] performed negative supervision on the embedding level by domain adaptation, thus can avoid negative sampling.

Though widely explored, the effectiveness of high-level visual features in this task is neglected. In this paper, we leverage the aesthetic features (additionally with semantic features and collaborative information) in the learning to rank process. For each positive item, we regard items with similar visual features or items connected in the bipartite graph as the neighbors (potential positive samples), and assume that users will prefer them to other negative samples.

## 3 Preliminaries

In this section, we introduce some preliminaries about the aesthetic neural network, which is used to extract the aesthetic features of clothing images.

Wang et al. [39] introduced the Brain-inspired Deep Networks (BDN, shown in Fig. 2), a deep CNN structure consists of several parallel pathways (sub-networks) and a high-level synthesis network. It is trained on the *Aesthetic Visual Analysis (AVA)* dataset, which contains 250,000 images with aesthetic ratings and tagged with 14 photographic styles (e.g., complementary colors, duotones, rule of thirds, etc.). The pathways take the form of convolutional networks to exact the abstractive aesthetic features by *pre-trained* with the individual labels of each tag. For example, when training the pathway for complementary colors, the individual label is 1 if the sample is tagged with "complementary colors" and is 0 if not.

We input the raw features, including low-level features (hue, saturation, value) and abstractive features (feature maps of the pathways), into the high-level synthesis network and *jointly tune* it with the pathways for aesthetic rating prediction. Considering that the *AVA* is a photography dataset and the styles are for photography, so not all the raw features extracted by the pathways are desired in our recommendation task, thus we only reserve the pathways that are relevant to the clothing aesthetic. Finally, we use the output of the second fully connected layer of the synthesis network as our aesthetic features.

We then analyze several extensively used features to illustrate the superiority of our aesthetic features.

**CNN Features:** These are the most extensively used features due to their extraordinary representation ability. Trained for the image classification task, CNN extracts the features important to image semantics, thus CNN features mainly contain semantic information, which contributes little to evaluate the aesthetics of an image. Recall the example in Fig. 1, it can encode "There is a skirt in the image" but cannot express "The clothing is beautiful and fits the user's taste." Devised for aesthetic assessment, BDN can capture the high-level aesthetic information. As such, our aesthetic features can do better in beauty estimating and complement CNN features in clothing recommendation.

**Color Histograms:** Zhao et al. [46] exploited color histograms to represent human's feeling about the posters and frames for movie recommendation. Though can get the aesthetic information roughly, the low-level handcrafted features are crude, unilateral, and empirical. BDN can get abun-

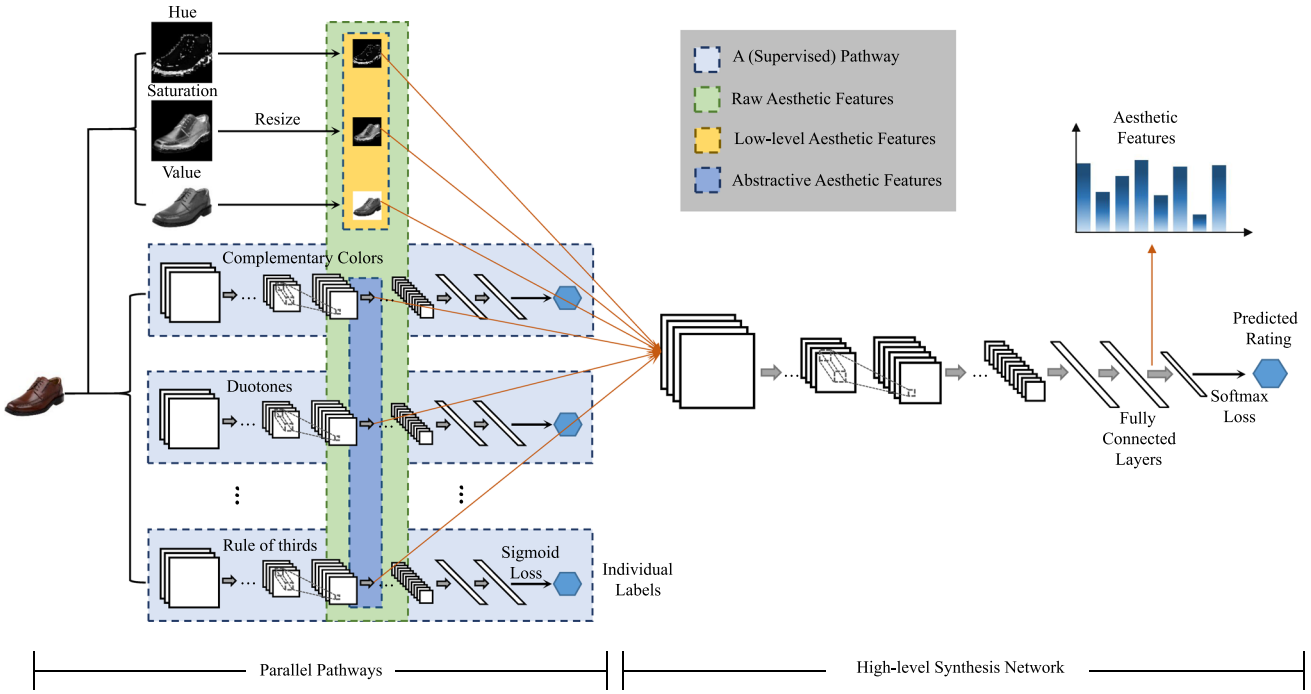



**Fig. 2** Brain-inspired deep network (BDN) architecture

dant visual features by the pathways. Also, it is data-driven, since the rules to extract features are learned from the data. Compared with the intuitive color histograms, our aesthetic features are more objective and comprehensive. Recall the example in Fig. 1 again, color histograms can tell us no more than "The clothes in the image is white and black."

## 4 Aesthetic-based recommendation

In this section, we first introduce the basic tensor factorization model, and then integrate visual features into the basic model to propose the visually aware recommendation with aesthetic features (VRA) model. The summary of notations is represented in Table 1.

### 4.1 Basic model

Considering the impact of time on aesthetic preferences, we propose a context-aware model as the basic model to account for the temporal factor. We use a $P \times Q \times R$ tensor $\mathbf{A}$ to indicate the purchase events among the user, clothes, and time dimensions (where $P$, $Q$, $R$ are the number of users, clothes, and time intervals, respectively). If user $p$ purchased item $q$ in time interval $r$, $\mathbf{A}_{pqr} = 1$, otherwise $\mathbf{A}_{pqr} = 0$. Tensor factorization has been widely used to predict the missing entries (i.e., zero elements) in $\mathbf{A}$, which can be used for recommendation.

#### 4.1.1 Existing methods and their limitations

In this subsection, we summarize the motivation of our novel tensor factorization model by revealing the limitations of existing models.

**Tucker Decomposition:** This method [21] decomposes the tensor $\mathbf{A}$ into a tensor core and three matrices,

$$\hat{\mathbf{A}}_{pqr} = \sum_{i=1}^{K_1}\sum_{j=1}^{K_2}\sum_{k=1}^{K_3} \mathbf{a}_{ijk}\mathbf{U}_{ip}\mathbf{V}_{jq}\mathbf{T}_{kr},$$

where $\mathbf{a} \in \mathbb{R}^{K_1 \times K_2 \times K_3}$ is the tensor core, $\mathbf{U} \in \mathbb{R}^{K_1 \times P}$, $\mathbf{V} \in \mathbb{R}^{K_2 \times Q}$, and $\mathbf{T} \in \mathbb{R}^{K_3 \times R}$. Tucker decomposition has very strong representation ability, but it is very time consuming, and hard to converge.

**CP Decomposition:** The tensor $\mathbf{A}$ is decomposed into three matrices in CP decomposition,

$$\hat{\mathbf{A}}_{pqr} = \sum_{k=1}^{K} \mathbf{U}_{kp}\mathbf{V}_{kq}\mathbf{T}_{kr},$$

where $\mathbf{U} \in \mathbb{R}^{K \times P}$, $\mathbf{V} \in \mathbb{R}^{K \times Q}$, and $\mathbf{T} \in \mathbb{R}^{K \times R}$. This model has been widely used due to its linear time complexity, especially in Coupled Matrix and Tensor Factorization (CMTF) structure models [1–3]. However, all dimensions (users, clothes, time) are mapped to the same latent factor space. Intuitively, we want the latent factors relating users and clothes to encode the information about users' prefer-

**Table 1** The summary of notations

| Notations | Definitions |
|---|---|
| $p/q/r$ | The $p$th user/$q$th item/$r$th time |
| $P/Q/R$ | The total number of users/items/time intervals |
| **A/B/C** | User–item–time tensor/user–item matrix/time–item matrix |
| $\hat{\mathbf{A}}/\hat{\mathbf{B}}/\hat{\mathbf{C}}$ | Reconstruction of **A/B/C** |
| $\mathbf{F}/\mathbf{f}_{CNN}/\mathbf{f}_{AES}$ | Visual features/CNN features/aesthetic features |
| $\mathbf{\Theta} = \{\mathbf{U}, \mathbf{V}, \mathbf{W}, \mathbf{T}, \mathbf{M}, \mathbf{N}\}$ | Model parameters |
| $\lambda_c/\lambda_r$ | The weighting parameter/regularization coefficient |
| $\mathcal{P}$ | The set of $P$ users $\{p_1, p_2, \ldots, p_P\}$ |
| $\mathcal{Q}$ | The set of $Q$ items $\{q_1, q_2, \ldots, q_Q\}$ |
| $\mathcal{R}$ | The set of $R$ items $\{r_1, r_2, \ldots, r_R\}$ |
| $\mathcal{Q}_p^+/\mathcal{Q}_p^-$ | The set of items purchased/not purchased by user $p$ |
| $\mathcal{Q}_r^+/\mathcal{Q}_r^-$ | The set of items purchased/not purchased in time $r$ |
| $\mathcal{Q}_{pr}^+/\mathcal{Q}_{pr}^-$ | The set of items purchased/not purchased by user $p$ and/or in time $r$ |
| $\mathcal{D}$ | The training set of (user, positive item, time) tuples |
| $\mathcal{N}_q^C$ | The neighbor set of item $q$ constructed based on CNN features |
| $\mathcal{N}_q^A$ | The neighbor set of item $q$ constructed based on aesthetic features |
| $\mathcal{N}_q^U$ | The neighbor set of item $q$ constructed based on CNN users |
| $\mathcal{N}_q^T$ | The neighbor set of item $q$ constructed based on time |
| $\mathcal{A} \setminus \mathcal{B}$ | The set of elements in $\mathcal{A}$ but not in $\mathcal{B}$ |
| $\lvert\mathcal{A}\rvert$ | The size of set $\mathcal{A}$ |

ence, like aesthetics, prices, quality, brands, etc., and the latent factors relating clothes and time to encode the information about the seasonal characteristics and fashion elements of clothes like colors, thickness, design, etc.

**PITF Decomposition:** The Pairwise Interaction Tensor Factorization (PITF) model [35] decomposes **A** into three pair of matrices,

$$\hat{\mathbf{A}}_{pqr} = \sum_{k=1}^{K} \mathbf{U}_{kp}^{\mathbf{V}} \mathbf{V}_{kq}^{\mathbf{U}} + \sum_{k=1}^{K} \mathbf{U}_{kp}^{\mathbf{T}} \mathbf{T}_{kr}^{\mathbf{U}} + \sum_{k=1}^{K} \mathbf{V}_{kq}^{\mathbf{T}} \mathbf{T}_{kr}^{\mathbf{V}},$$

where $\mathbf{U}^{\mathbf{V}}, \mathbf{U}^{\mathbf{T}} \in \mathbb{R}^{K \times P}$; $\mathbf{V}^{\mathbf{U}}, \mathbf{V}^{\mathbf{T}} \in \mathbb{R}^{K \times Q}$; $\mathbf{T}^{\mathbf{U}}, \mathbf{T}^{\mathbf{V}} \in \mathbb{R}^{K \times R}$. PIFT has a linear complexity and strong representation ability. Yet, it is not in line with implicit feedbacks due to the additive combination of each pair of matrices. For example, in PIFT, for certain clothes $q$ liked by the user $p$ but not fitting the current time $r$, $q$ gets a high score for $p$ and a low score for $r$. It should not be recommended to the user since we want to recommend the right item in the right time. However, the total score can be high enough if $p$ likes $q$ so much that $q$'s score for $p$ is very high. In this case, $q$ will be returned even it does not fit the time. In addition, PITF model is inappropriate to be trained with coupled matrices.

#### 4.1.2 Model formulation

To address the limitations of the aforementioned models, we propose a new tensor factorization method which is for implicit feedback with linear complexity. When a user makes a purchase decision on a clothing product, there are two primary factors: if the product fits the user's preferences and if it fits the time. A clothing product fits a user's preferences if the appearance is appealing, the style fits the user's tastes, the quality is good, and the price is acceptable. And a clothing product fits the time if it is in-season and fashionable. For user $p$, clothing $q$, and time interval $r$, we use the scores $S_1$ and $S_2$ to indicate how the user likes the clothing and how the clothing fits the time, respectively. $S_1 = 1$ when the user likes the clothing and $S_1 = 0$ otherwise. Similarly, $S_2 = 1$ if the clothing fits the time and $S_2 = 0$ otherwise. The user will buy the clothing only if $S_1 = 1$ and $S_2 = 1$, so, $\hat{\mathbf{A}}_{pqr} = S_1 \& S_2$. To make the formula differentiable, we can approximately formulate it as $\hat{\mathbf{A}}_{pqr} = S_1 \cdot S_2$. We present $S_1$ and $S_2$ in the form of matrix factorization: $S_1 = \mathbf{U}_{*p}^{\mathsf{T}} \mathbf{V}_{*q}$, $S_2 = \mathbf{T}_{*r}^{\mathsf{T}} \mathbf{W}_{*q}$, where $\mathbf{U} \in \mathbb{R}^{K_1 \times P}$, $\mathbf{V} \in \mathbb{R}^{K_1 \times Q}$, $\mathbf{T} \in \mathbb{R}^{K_2 \times R}$, and $\mathbf{W} \in \mathbb{R}^{K_2 \times Q}$. The prediction is then given by:

$$\hat{\mathbf{A}}_{pqr} = \left(\mathbf{U}_{*p}^{\mathsf{T}} \mathbf{V}_{*q}\right) \left(\mathbf{T}_{*r}^{\mathsf{T}} \mathbf{W}_{*q}\right). \tag{1}$$

We can see that in Eq. (1), the latent factors relating users and clothes are independent with those relating clothes and time. Though the $K_1$-dimensional vector $\mathbf{V}_{*q}$ and the $K_2$-dimensional vector $\mathbf{W}_{*q}$ are all latent factors of clothing $q$, $\mathbf{V}_{*q}$ captures the information about users' preferences whereas $\mathbf{W}_{*q}$ captures the temporal information of the clothing. Compared with CP decomposition, our model is more effective and expressive in capturing the underlying latent patterns in purchases. Compared with PITF, combining $S_1$ and $S_2$ with & (approximated by multiplication) is helpful to recommend right clothing in right time. Moreover, our model is efficient and easy to train compared with the Tucker decomposition.

#### 4.1.3 Coupled matrix and tensor factorization

Though widely used to portray the context information in recommendation, tensor factorization suffers from poor convergence due to the sparsity of the tensor. To relieve this problem, Acar et al. [2] proposed a CMTF model, which decomposes the tensor with coupled matrices. In this subsection, we couple our tensor factorization model with restrained matrices during training. As our model is proposed by considering two factors: $S_1$ (use's preference toward items) and $S_2$ (time's "preference" toward items), we also explore restrained matrices that can supervise these two factors.

**User $\times$ Clothing Matrix:** We use matrix $\mathbf{B} \in \mathbb{R}^{P\times Q}$ to indicate the purchase activities between users and clothes. $\mathbf{B}_{pq} = 1$ if user $p$ purchased clothing $q$ and $\mathbf{B}_{pq} = 0$ if not. We use $\mathbf{B}$ to supervise $S_1$ when learning our model.

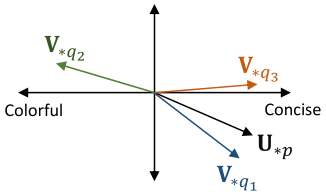
**Time $\times$ Clothing Matrix:** We use matrix $\mathbf{C} \in \mathbb{R}^{R\times Q}$ to record when the clothing was purchased. Since the characteristics of clothing change steadily with time, we make a coarse-grained discretization on time to avoid the tensor from being extremely sparse. Time is divided into $R$ intervals in total. $\mathbf{C}_{rq} = 1$ if clothing $q$ is purchased in time interval $r$ and $\mathbf{C}_{rq} = 0$ if not. We use $\mathbf{C}$ to supervise $S_2$.

In previous work [2,3,20,40], the CMTF models are optimized by minimizing the reconstruction loss (MSE_Opt):

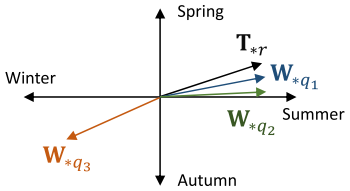
$$\text{MSE\_Opt}=\frac{1}{2}\left\|\mathbf{A}-\hat{\mathbf{A}}\right\|_{\mathrm{F}}^{2}+\frac{\lambda_{\mathrm{c}}}{2}\left(\left\|\mathbf{B}-\hat{\mathbf{B}}\right\|_{\mathrm{F}}^{2}+\left\|\mathbf{C}-\hat{\mathbf{C}}\right\|_{\mathrm{F}}^{2}\right)+\frac{\lambda_{\mathrm{r}}}{2}\|\boldsymbol{\Theta}\|_{\mathrm{F}}^{2} \tag{2}$$

where $\hat{\mathbf{A}}$, $\hat{\mathbf{B}}$, and $\hat{\mathbf{C}}$ are the reconstructions of $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{C}$, respectively. $\hat{\mathbf{A}}$ is defined in Eq. (1), $\hat{\mathbf{B}} = \mathbf{U}^{\mathrm{T}}\mathbf{V}$, and $\hat{\mathbf{C}} = \mathbf{T}^{\mathrm{T}}\mathbf{W}$; $\lambda_c$ is a parameter to balance the weights of the tensor term and coupled matrix terms. The last term of Eq. (2) is the regularization term to prevent overfitting, and $\lambda_r$ is the regularization coefficient. $\|\ \|_{\mathrm{F}}$ is the Frobenius norm of a matrix, $\boldsymbol{\Theta}$ represents the parameters of the model, $\boldsymbol{\Theta} = \{\mathbf{U}, \mathbf{V}, \mathbf{T}, \mathbf{W}\}$. As shown in Eq. (2), we train model parameters to complete $\mathbf{A}$, and use $\mathbf{B}$ and $\mathbf{C}$ to assist the supervision of model training.

(a) Item examples



(b) User-item latent space



(c) Time-item latent space

**Fig. 3** An example to illustrate our basic model

***Example 1*** We give an example to illustrate how our basic model works (please see Fig. 3). There are three items ($q_1$, $q_2$, and $q_3$) and two latent factor spaces (the user–item latent space and time–item latent space). The user–item latent space encodes the users' preference and the time–item latent space encodes the temporal characteristics of items. In our basic model, we map users and items into user–item latent space by $\mathbf{U}$ and $\mathbf{V}$, and map time intervals and items into time–item latent space by $\mathbf{T}$ and $\mathbf{W}$. In this example, we aim to recommend clothes to a user $p$ who likes simple and elegant clothes in a time interval $r$ in summer. For clothing $q_1$, we can see that it fits $p$'s preference and it is a shirt designed for summer, thus $q_1$ gets high $S_1$ and $S_2$ scores and can be recommended due to the high score $S = S_1 \cdot S_2$. For the clothing $q_2$, it is a piece of summer clothes yet is too colorful for $p$, thus $q_2$ gets low $S_1$ score and high $S_2$ score and cannot be recommended. Clothing $q_3$ is simple and elegant yet is used in winter, thus $q_3$ gets high $S_1$ score and low $S_2$ score and cannot be recommended either.

If we neglect the first term in Eq. (2), predicting $S_1$ in the user–item latent space and predicting $S_2$ in the time–item space are two independent recommendation tasks. Supervised by $\mathbf{B}$, predicting $S_1$ is a conventional recommendation task which recommends items to users. Supervised by $\mathbf{C}$, pre-

dicting $S_2$ is to "recommend" items to current time. When predicting $S_2$, we need to encode "preferences" of time in the time–item latent space. These "preferences" may relate to the seasonal or fashion information.

### 4.2 Hybrid model

In this section, we incorporate the visual features into the basic model, and optimize it with the pairwise learning to rank method.

#### 4.2.1 Model formulation

Combined with visual features, we formulate the predictive model as:

$$\hat{\mathbf{A}}_{pqr} = \left(\mathbf{U}_{*p}^{\mathsf{T}}\mathbf{V}_{*q} + \mathbf{M}_{*p}^{\mathsf{T}}\mathbf{F}_{*q}\right)\left(\mathbf{T}_{*r}^{\mathsf{T}}\mathbf{W}_{*q} + \mathbf{N}_{*r}^{\mathsf{T}}\mathbf{F}_{*q}\right), \quad (3)$$

where $\mathbf{F} \in \mathbb{R}^{2K\times Q}$ is the feature matrix, $\mathbf{F}_{*q}$ is the visual features of clothing $q$, which is the concatenation of CNN features ($\mathbf{f}_{CNN} \in \mathbb{R}^{K\times 1}$) and aesthetic features ($\mathbf{f}_{AES} \in \mathbb{R}^{K\times 1}$), $\mathbf{F}_{*q} = \begin{bmatrix}\mathbf{f}_{CNN}\\ \mathbf{f}_{AES}\end{bmatrix}$ and $K = 4096$. $\mathbf{M} \in \mathbb{R}^{2K\times P}$ and $\mathbf{N} \in \mathbb{R}^{2K\times R}$ are visual preference matrices. $\mathbf{M}_{*p}$ encodes the visual preferences of user $p$ and $\mathbf{N}_{*r}$ encodes the visual preferences in time interval $r$. In our model, both the latent factors and visual features contribute to the final prediction. Though the latent factors can uncover any relevant attributes theoretically, they usually cannot in real-world applications on account of the sparsity of the data and lack of information. So the assistance of visual information can highly enhance the model. Also, recommender systems often suffer from the *cold start* problem. We cannot extract information for users and clothes without consumption records in CF methods. In this case, extra (visual and context) information can alleviate this problem. For example, for certain "cold" clothing $q$, we can decide whether to recommend it to a certain user $p$ in current time $r$ according to if $q$ looks satisfying to the user (determined by $\mathbf{M}_{*p}$) and to the time (determined by $\mathbf{N}_{*r}$). The model structure is illustrated in Fig. 4.

#### 4.2.2 Pairwise learning to rank

Since the Mean Squared Error Optimization (MSE_Opt, please see Eq. (2)), which is widely used in existing CMTF models [2,3,20,40], is designed for explicit feedback data, we design **P**airwise **L**earning to **R**ank (**PLR**) method with coupled matrix constrain for our VRA on implicit feedback data. We represent the positive set $\mathcal{D}$ in the form of triples:

$$\mathcal{D} = \left\{(p,q,r)\middle|\mathbf{A}_{pqr} = 1\right\},$$

and the set of unlabeled samples is:

$$\mathcal{Q}_{pr}^{-} = \left\{q\middle|q \in \mathcal{Q} \setminus \left(\mathcal{Q}_p^{+} \cup \mathcal{Q}_r^{+}\right)\right\},$$

where $\mathcal{Q}$ denotes the set of items, $\mathcal{Q}_p^{+} = \left\{q\middle|\mathbf{B}_{pq} = 1\right\}$ denotes the set of items purchased by user $p$, and $\mathcal{Q}_r^{+} = \left\{q\middle|\mathbf{C}_{rq} = 1\right\}$ denotes the set of items purchased in time $r$. The objective function is formulated as:

$$\text{PLR\_Opt} = \sum_{(p,q,r)\in\mathcal{D}} \sum_{q'\in\mathcal{Q}_{pr}^{-}} L\left(p,q,q',r\right) - \frac{\lambda_r}{2}\|\boldsymbol{\Theta}\|_{\mathrm{F}}^2. \quad (4)$$

$L(\,)$ in Eq. (4) is the likelihood function,

$$L\left(p,q,q',r\right) = \ln\sigma\left(\hat{\mathbf{A}}_{pqq'r}\right) + \lambda_c \left[\ln\sigma\left(\hat{\mathbf{B}}_{pqq'}\right) + \ln\sigma\left(\hat{\mathbf{C}}_{rqq'}\right)\right],$$

where $\hat{\mathbf{A}}$ is defined in Eq. (3), $\hat{\mathbf{B}} = \mathbf{U}^{\mathsf{T}}\mathbf{V} + \mathbf{M}^{\mathsf{T}}\mathbf{F}$, and $\hat{\mathbf{C}} = \mathbf{T}^{\mathsf{T}}\mathbf{W} + \mathbf{N}^{\mathsf{T}}\mathbf{F}$; $\hat{\mathbf{A}}_{pqq'r} = \hat{\mathbf{A}}_{pqr} - \hat{\mathbf{A}}_{pq'r}$, $\hat{\mathbf{B}}_{pqq'} = \hat{\mathbf{B}}_{pq} - \hat{\mathbf{B}}_{pq'}$, $\hat{\mathbf{C}}_{rqq'} = \hat{\mathbf{C}}_{rq} - \hat{\mathbf{C}}_{rq'}$; $\sigma(\,)$ is the sigmoid function; The model is optimized from users' *implicit feedback* with mini-batch gradient descent, which calculates the gradient with a small batch of samples.

## 5 Aesthetic-enhanced pairwise learning to rank

In Sect. 4, we leverage aesthetic features to model users' aesthetic preference and in this section, we use aesthetic features to improve ranking performance (CNN features and collaborative information are leveraged as well for comprehensiveness). PLR which is introduced in Sect. 4.2.2 is a pairwise learning method for multi-objective optimization with the aim of maximizing the gap between the positive feedbacks and negative feedbacks. Pairwise learning has been widely used due to its strong performance [6,8,14] while there is a critical issue in the current formulation. To be specific, a user did not purchase a product may because she is not interested in it, but may also because that she has never seen it before. However, in pairwise learning, all missing entries are treated as negative samples hence many potential positive samples are mislabeled as negative ones. To uncover these potential positive samples, we construct the neighbor set $\mathcal{N}_q$ for each positive sample $q$ by uncovering the products that have similar visual representations with $q$, or the products connected to $q$ in the user–item or time–item graphs. In other words, $\mathcal{N}_q$ contains the products near $q$ in the visual space or in the graphs. If a user purchased $q$, she may also prefer $\mathcal{N}_q$ due to the similarity on comprehensive aspects. In this

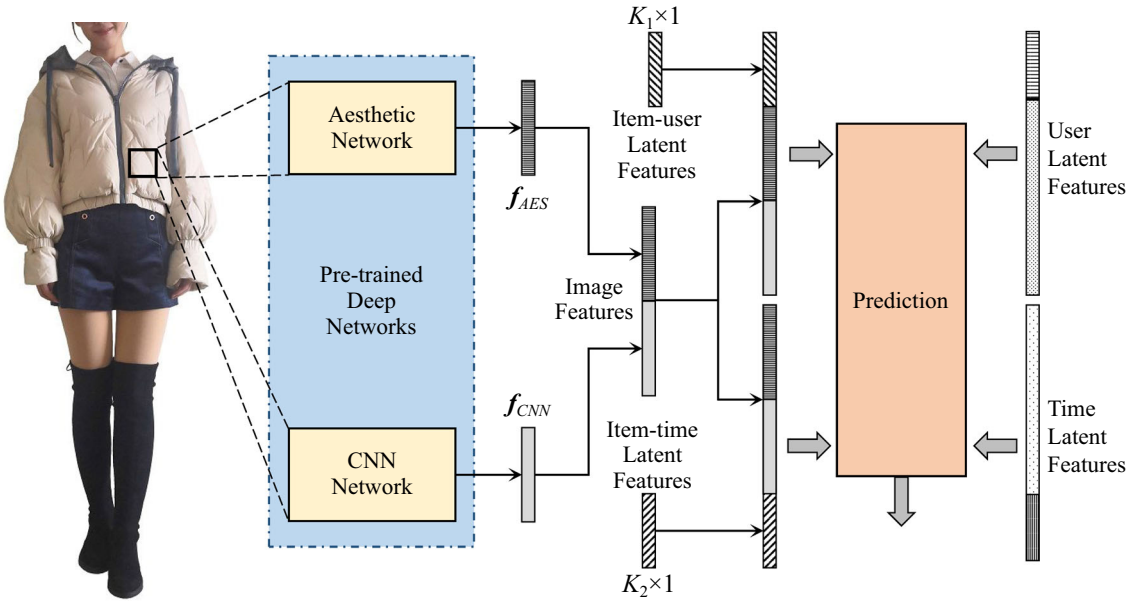



**Fig. 4** Diagram of our preference predictor

section, we propose an Aesthetic-enhanced Pairwise Learning to Rank (APLR) by considering these potential positive samples in ranking.

### 5.1 Problem formulation

When sampling, we regard the neighbors as potential positive samples. For a user $p$ and a time interval $r$, we assume that (1) user $p$ prefers items with positive feedbacks to the others; (2) user $p$ prefers the neighbors of the positive sample to the irrelevant ones; (3) positive samples fit the current time $r$ better than the others; (4) neighbors of the positive sample fit the current time $r$ better than the irrelevant ones. So for each $(p, q, r)$ in $\mathcal{D}$, we have the preference relationship,

$$(p,q,r) \succ (p, \mathcal{Q}_{pr}^{-}, r), (p,q,r) \succ (p, \mathcal{N}_q, r),$$
$$(p, \mathcal{N}_q, r) \succ (p, \mathcal{Q}_{pr}^{-} \setminus \mathcal{N}_q, r).$$

As such, we can generalize Eq. (4) as follows:

$$\begin{aligned} \text{APLR\_OPT} &= \sum_{(p,q,r)\in\mathcal{D}} \Bigg[ \sum_{q''\in\mathcal{Q}_{pr}^{-}} L\left(p,q,q'',r\right) + \eta_1 \sum_{q'\in\mathcal{N}_q} L\left(p,q,q',r\right) \\ &\quad + \eta_2 \sum_{q'\in\mathcal{N}_q} \sum_{q''\in\mathcal{Q}_{pr}^{-}\setminus\mathcal{N}_q} L\left(p,q',q'',r\right) \Bigg] - \frac{\lambda_r}{2}\|\boldsymbol{\Theta}\|_{\mathrm{F}}^2, \end{aligned} \tag{5}$$

where $\eta_1$ and $\eta_2$ are weighting parameters. Here, we can see that for each purchase record $(p, q, r)$, user $p$ prefers $q$ to $q'$ and prefers $q'$ to $q''$. The preference relationship is constructed by finding the neighbors of the positive items, which can be interpreted as an item-based collaborative learning model [17]. Most existing works learn to rank by constructing the potential set of each user [11,23,32–34,47]. In the next subsections, we will introduce how to construct neighbor set for each item, and demonstrate the advantages of our item-based collaborative sampling strategy.

### 5.2 Constructing neighbor set

To find the neighbors of each positive sample, we leverage the visual information and the collaborative information. For visual information, we cluster all products with CNN features and aesthetic features. For each product, the cluster it belongs to is the neighborhood set. And for collaborative information, we find all products purchased by the same user or purchased in the same time to be the neighbor products.

**Neighbors in aesthetic space:** Similarly, we cluster all products by the aesthetic features and regard the cluster a product $q$ belongs to as the aesthetic neighbor set, denoted as $\mathcal{N}_q^A$. Products close to each other in the aesthetic space have similar aesthetic characteristics. For a certain user, since that positive samples are in line with her aesthetics, neighbors are also in line with her aesthetics.

**Neighbors in semantic space:** We cluster all products by the CNN features. For a product $q$, the cluster it belongs to is the semantic neighbor set, denoted as $\mathcal{N}_q^C$. Products with similar CNN features have similar appearances, users may have interests in the items that look like the purchased ones.

**Neighbors linked by users:** For each product $q$, we find all products that purchased by the same user to consist the

user-linked neighbor set, $\mathcal{N}_q^U = \{q' | \mathbf{B}_{pq} = 1 \wedge \mathbf{B}_{pq'} = 1\}$. Each product $q'$ in $\mathcal{N}_q^U$ has been purchased by the same user with $q$, therefore users who have interests in $q$ may also like $q'$. We update the part of our model which captures the users' preferences (parameters $\mathbf{U}$, $\mathbf{V}$, and $\mathbf{M}$) with $\mathcal{N}_q^U$.

**Neighbors linked by time:** For each product $q$, we find all products that purchased in the same time with $q$ to consist the time-linked neighbor set, $\mathcal{N}_q^T = \{q' | \mathbf{C}_{rq} = 1 \wedge \mathbf{C}_{rq'} = 1\}$. Each product $q'$ in $\mathcal{N}_q^T$ has been purchased in the same time with the current product $q$, so $q'$ may fit the current time better than other missing value samples. We update the part which captures the temporal character of products in our model (parameters $\mathbf{T}$, $\mathbf{W}$, and $\mathbf{N}$) with $\mathcal{N}_q^T$.

$\mathcal{N}_q^U$ is the neighbor set of $q$ in user–item bipartite graph and $\mathcal{N}_q^T$ is that in time–item bipartite graph. Of special notice is that they are used to update different parts of our model. Taking $\mathcal{N}_q^T$ as an example, it only contributes to predicting $S_2$. As we discussed in Example 1, when predicting $S_2$, we recommend items to each time interval, i.e., we capture the "preference" of current time $r$ rather than of current user $p$, and return "personalized" recommendation to $r$. In this situation, two items $q_1$ and $q_2$ that both fit $r$ are two similar items from time perspective, though they may be totally different when considering the preference of the user, therefore we only use $\mathcal{N}_q^T$ to update $\{\mathbf{T}, \mathbf{W}, \mathbf{N}\}$ rather than $\{\mathbf{U}, \mathbf{V}, \mathbf{M}\}$.

$\mathcal{N}_q^T$ is an extension of $\mathcal{N}_q^U$ from the user–item graph to time–item graph. Considering the difference between user $p$ and time $r$ ($p$ is an index and $r$ is a discrete numerical value), a more general way to construct $\mathcal{N}_q^T$ is to set a window $\Delta r$, and for each product $q$, $\mathcal{N}_q^T$ contains all products that purchased in similar time (in range of $r \pm \Delta r$) with $q$, i.e., $\mathcal{N}_q^T = \{q' | \mathbf{C}_{rq} = 1 \wedge \mathbf{C}_{r'q'} = 1 \wedge |r - r'| \leq \Delta r\}$. Since the density of $\mathbf{C}$ is much higher than $\mathbf{B}$, the size of $\mathcal{N}_q^T$ will be very large when we set a large $\Delta r$, we set $\Delta r = 0$ in our APLR.

### 5.3 Model learning

We then calculate the gradient of Eq. (5). To maximize the objective function, we take the first-order derivatives with respect to each model parameter:

$$\begin{aligned}\nabla_{\mathbf{\Theta}}\text{APLR\_OPT} &= \sum_{(p,q,r)\in\mathcal{D}} \Bigg[ \sum_{q''\in\mathcal{Q}_{pr}^-} \frac{\partial L(p,q,q'',r)}{\partial \mathbf{\Theta}} + \eta_1 \sum_{q'\in\mathcal{N}_q} \frac{\partial L(p,q,q',r)}{\partial \mathbf{\Theta}} \\ &\quad + \eta_2 \sum_{q'\in\mathcal{N}_q} \sum_{q''\in\mathcal{Q}_{pr}^- \setminus \mathcal{N}_q} \frac{\partial L(p,q',q'',r)}{\partial \mathbf{\Theta}} \Bigg] - \lambda_r \mathbf{\Theta}, \end{aligned} \tag{6}$$

where

$$\begin{aligned}\frac{\partial L(p,q,q',r)}{\partial \mathbf{\Theta}} =& \sigma\left(-\hat{\mathbf{A}}_{pqq'r}\right)\frac{\partial \hat{\mathbf{A}}_{pqq'r}}{\partial \mathbf{\Theta}} \\ &+ \lambda_c\left[\sigma\left(-\hat{\mathbf{B}}_{pqq'}\right)\frac{\partial \hat{\mathbf{B}}_{pqq'}}{\partial \mathbf{\Theta}}\right. \\ &\left. + \sigma\left(-\hat{\mathbf{C}}_{rqq'}\right)\frac{\partial \hat{\mathbf{C}}_{rqq'}}{\partial \mathbf{\Theta}}\right].\end{aligned}$$

We use $\boldsymbol{\theta}$ to denote certain column of $\mathbf{\Theta}$. For our VRA model, the derivatives are:

$$\frac{\partial \hat{\mathbf{A}}_{pqq'r}}{\partial \boldsymbol{\theta}} = \begin{cases} \hat{\mathbf{C}}_{rq}\mathbf{V}_{*q} - \hat{\mathbf{C}}_{rq'}\mathbf{V}_{*q'} & \text{if } \boldsymbol{\theta} = \mathbf{U}_{*p} \\ \hat{\mathbf{C}}_{rq}\mathbf{U}_{*p}/ - \hat{\mathbf{C}}_{rq'}\mathbf{U}_{*p} & \text{if } \boldsymbol{\theta} = \mathbf{V}_{*q}/\mathbf{V}_{*q'} \\ \hat{\mathbf{C}}_{rq}\mathbf{F}_{*q} - \hat{\mathbf{C}}_{rq'}\mathbf{F}_{*q'} & \text{if } \boldsymbol{\theta} = \mathbf{M}_{*p} \end{cases} \tag{7}$$

$$\frac{\partial \hat{\mathbf{B}}_{pqq'}}{\partial \boldsymbol{\theta}} = \begin{cases} \mathbf{V}_{*q} - \mathbf{V}_{*q'} & \text{if } \boldsymbol{\theta} = \mathbf{U}_{*p} \\ \mathbf{U}_{*p}/ - \mathbf{U}_{*p} & \text{if } \boldsymbol{\theta} = \mathbf{V}_{*q}/\mathbf{V}_{*q'} \\ \mathbf{F}_{*q} - \mathbf{F}_{*q'} & \text{if } \boldsymbol{\theta} = \mathbf{M}_{*p} \end{cases}. \tag{8}$$

Eqs. (7) and (8) give the derivatives for $\mathbf{\Theta} = \{\mathbf{U}, \mathbf{V}, \mathbf{M}\}$, and we can easily get the same form for $\mathbf{\Theta} = \{\mathbf{T}, \mathbf{W}, \mathbf{N}\}$. $\frac{\partial \hat{\mathbf{A}}_{pqq'r}}{\partial \boldsymbol{\theta}}$ in Eq. (7) is certain column of $\frac{\partial \hat{\mathbf{A}}_{pqq'r}}{\partial \mathbf{\Theta}}$ in Eq. (6), for example, the $p$th column when $\boldsymbol{\theta} = \mathbf{U}_{*p}$.

Finally, we update the parameters with the derivatives we get. As discussed in Sect. 5.2, we use different neighborhood sets to update different parts of the model. For $\mathbf{\Theta} = \{\mathbf{U}, \mathbf{V}, \mathbf{M}\}$, we update the parameters:

$$\mathbf{\Theta} = \mathbf{\Theta} + \eta \nabla_{\mathbf{\Theta}} \text{APLR\_OPT}\big|_{\mathcal{N}_q = \mathcal{N}_q^U \cup \mathcal{N}_q^C \cup \mathcal{N}_q^A},$$

and for $\mathbf{\Theta} = \{\mathbf{T}, \mathbf{W}, \mathbf{N}\}$,

$$\mathbf{\Theta} = \mathbf{\Theta} + \eta \nabla_{\mathbf{\Theta}} \text{APLR\_OPT}\big|_{\mathcal{N}_q = \mathcal{N}_q^T \cup \mathcal{N}_q^C \cup \mathcal{N}_q^A}.$$

Our model is optimized with mini-batch gradient descent and for each positive sample, we sample $\rho$ negative samples and $\rho$ neighbors randomly to construct pairs, where $\rho$ is the sampling rate.

The detailed learning procedures about our method are shown in Algorithm 1. We first construct $\mathcal{N}_q^C$ and $\mathcal{N}_q^A$ by clustering the CNN features and aesthetic features, and construct $\mathcal{N}_q^U$ and $\mathcal{N}_q^T$ by collecting neighbors in user–item and time–item graph (line 1). We then exploit the mini-batch gradient descent to maximize the objective function. For each iteration, all positive samples are enumerated (lines 4–19). We compute the gradients with a batch containing $b$ positive samples (line 6), and select $\rho$ neighbors and $\rho$ negative samples (lines 8–16) construct preference pairs. Different parts of the model are updated with different samples (lines 12, 16, 17, and 18). To calculate the gradients (line 10), we combine Eq. (6) with Eqs. (7) and (8). One thing needs to be point

**Algorithm 1:** Learning VRA by APLR.

**Input**: sparse tensor $\mathbf{A}$, coupled matrices $\mathbf{B}$ and $\mathbf{C}$, visual features $\mathbf{F}$, weight coefficient for coupled matrices $\lambda_c$, regularization coefficient $\lambda_r$, weighting parameters $\eta_1$ and $\eta_2$, batch size $b$, learning rate $\eta$, sample rate $\rho$, maximum number of iterations $iter\_max$, and convergence criteria.

**Output**: top-$n$ prediction given by the complete tensor $\hat{\mathbf{A}}$.

construct $\mathcal{N}_q^C$, $\mathcal{N}_q^A$, $\mathcal{N}_q^U$, and $\mathcal{N}_q^T$ for each item $q$;
initialize $\boldsymbol{\Theta}$ randomly;
$iter = 0$;
**while** *not converged* && $iter < iter\_max$ **do**
  $iter += 1$;
  split all purchase records into $b$-size batches;
  **for** *each batch* **do**
    **for** *each record in current batch* **do**
      $\mathcal{N}_q = \mathcal{N}_q^C \bigcup \mathcal{N}_q^A \bigcup \mathcal{N}_q^U$;
      select $\rho$ neighbor items $q'$ randomly from $\mathcal{N}_q$;
      select $\rho$ neighbor items $q''$ randomly from $\mathcal{Q}_{pr}^- \setminus \mathcal{N}_q$;
      calculate and accumulate $\nabla_{\{\mathbf{U},\mathbf{V},\mathbf{M}\}}$APLR_OPT;
      $\mathcal{N}_q = \mathcal{N}_q^C \bigcup \mathcal{N}_q^A \bigcup \mathcal{N}_q^T$;
      select $\rho$ neighbor items $q'$ randomly from $\mathcal{N}_q$;
      select $\rho$ neighbor items $q''$ randomly from $\mathcal{Q}_{pr}^- \setminus \mathcal{N}_q$;
      calculate and accumulate $\nabla_{\{\mathbf{T},\mathbf{W},\mathbf{N}\}}$APLR_OPT;
    $\{\mathbf{U}, \mathbf{V}, \mathbf{M}\} += \eta \nabla_{\{\mathbf{U},\mathbf{V},\mathbf{M}\}}$BPR_OPT;
    $\{\mathbf{T}, \mathbf{W}, \mathbf{N}\} += \eta \nabla_{\{\mathbf{T},\mathbf{W},\mathbf{N}\}}$BPR_OPT;
  calculate $\hat{\mathbf{A}}$ and predict the top-$n$ items;
**return** *the top-n items*;

out is that $\frac{\partial \hat{\mathbf{A}}_{pqq'r}}{\partial \boldsymbol{\Theta}}$ in Eq. (7) is a certain column of $\frac{\partial \hat{\mathbf{A}}_{pqq'r}}{\partial \boldsymbol{\theta}}$ in Eq. (6), for example, the $p$th column when $\boldsymbol{\theta} = \mathbf{U}_{*p}$.

As we know, a more popular item tells us less about user's preference, and our optimization can weaken the contribution of popular items by nature. For a popular item $q_1$ and a minority item $q_2$, $|\mathcal{N}_{q_1}| \gg |\mathcal{N}_{q_2}|$, where $|\ \ |$ is the set size. Noting that a popular item connects to a large proportion of items, $\mathcal{N}_{q_1}$ contains various items and depicts little about the preference, and we only select a small proportion of $\mathcal{N}_{q_1}$ ($\frac{\rho}{|\mathcal{N}_{q_1}|}$). For a minority item $q_2$, $\mathcal{N}_{q_2}$ only contains similar items which the current user may prefer, therefore we select a large proportion of $\mathcal{N}_{q_2}$ ($\frac{\rho}{|\mathcal{N}_{q_2}|}$). If $q_2$ is very unpopular, we can almost cover $\mathcal{N}_{q_2}$ by sampling $\rho$ samples.

Another advantage of our neighbor-enhanced pairwise optimization is that important neighbors can be strengthened in the probability level. We give an example to illustrate this advantage.

***Example 2*** As shown in Fig. 5, $p$ is the current user, and $q_1$, $q_2$, $q_3$ are three positive samples. $q_{4-10}$ are neighbors of positive samples and with high probability to be preferred by $p$, hence are potential positive samples. As we can see, $q_4$ is the most important potential items since it is the neighbor of all $p$'s purchased items and with the highest probability to be preferred. $q_{7-10}$ are not important since they are the neighbors of $q_3$, which is a popular item. As we discussed,

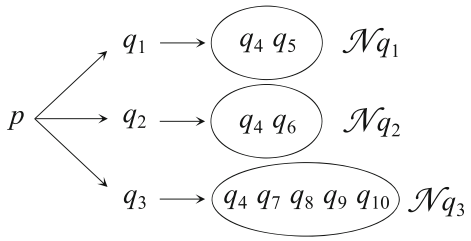



**Fig. 5** An example of neighbor sets

$\mathcal{N}_{q_3}$ provides little information about $p$'s preference. When sampling from the neighbor set, taking $\rho = 1$ as an example, $q_4$ has $\frac{1}{20}$, $\frac{3}{10}$, $\frac{9}{20}$, and $\frac{1}{5}$ probability to be sampled 3 times, twice, once, and not to be sampled, respectively. $q_5$ and $q_6$ both have $\frac{1}{2}$ probability to be sampled (once) and have $\frac{1}{2}$ probability not to be sampled. $q_{7-10}$ all have $\frac{1}{5}$ probability to be sampled (once) and have $\frac{4}{5}$ probability not to be sampled. Assuming we iterate 200 times to train our model, $q_4$ can be sampled about 240 times, $q_5$ and $q_6$ can both be sampled about 100 times, and $q_{7-10}$ can all be sampled about 40 times. We can see that potential samples are weighted based on the importance in the probability (frequency) level. To improve the sampling quality, Liu et al. [23] weighted potential samples based on the strength of the connection yet additional computation is required. Compared with [23], our method weights potential samples by nature.

## 6 Experiments

In this section, we conduct experiments on real-world datasets to verify the effectiveness of our method. We focus on answering the following four key research questions:
**RQ1:** What factors affect users' aesthetics?
**RQ2:** How is the performance of our overall solution for the clothing recommendation task?
**RQ3:** How is the effectiveness of the aesthetic features compared with conventional visual features?
**RQ4:** How is the performance of our aesthetic-enhanced learning to rank method?

### 6.1 Experimental setup

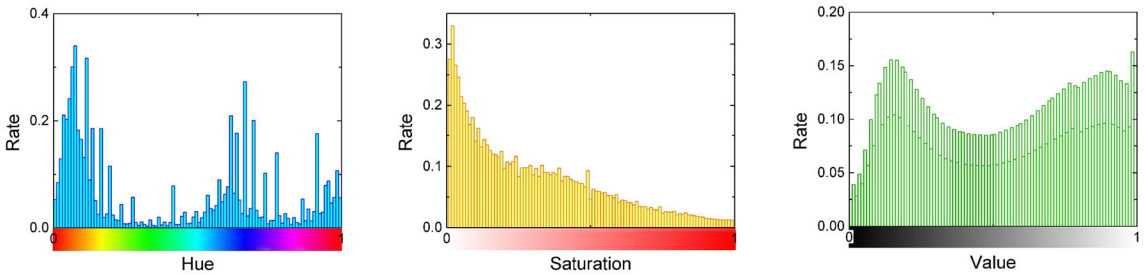
#### 6.1.1 Datasets

We use the *AVA* dataset to train the aesthetic network and use the *Amazon* dataset to train the recommendation models.

– **Aesthetic Visual Analysis (AVA):** We train the aesthetic network with the *AVA* dataset [30], which is the collection of images and meta-data derived from *DPChallenge.com*. It contains over 250,000 images with aesthetic ratings

**Table 2** Statistics of datasets

| Dataset | Purchase | User | Item | Sparsity of matrices/tensors |
|---|---|---|---|---|
| *Amazon* | 275539 | 39371 | 23022 | 99.9696%/99.9999% |
| *Men* | 67156 | 22547 | 5460 | 99.9454%/99.9998% |
| *Women* | 176136 | 35059 | 14500 | 99.9653%/99.9999% |
| *Clothes* | 115841 | 32728 | 8777 | 99.9597%/99.9998% |
| *Shoes* | 94560 | 32538 | 8231 | 99.9647%/99.9999% |
| *Jewelry* | 37314 | 15924 | 3607 | 99.9350%/99.9997% |




(a) Hue (b) Saturation (c) Value

**Fig. 6** Distribution of hue, saturation, and value of the whole dataset

from 1 to 10, 66 textual tags describing the semantics of images, and 14 photographic styles: complementary colors, duotones, negative image, rule of thirds, image grain, silhouettes, vanishing point, high dynamic range, light on white, long exposure, macro, motion blur, shallow DOF, and soft focus. We abandon the last 7 styles when constructing pathways in our aesthetic feature extractor since they are about camera setting.

– **Amazon:** The *Amazon* dataset [14] is the consumption records from *Amazon.com*. In this paper, we use the *clothing shoes and jewelry* category filtered with *5-core* (remove users and items with less than 5 purchase records) to train all recommendation models. Please note that in the below part of this paper, we use *Amazon* to denote the *clothing shoes and jewelry* category.

#### 6.1.2 Experiment settings

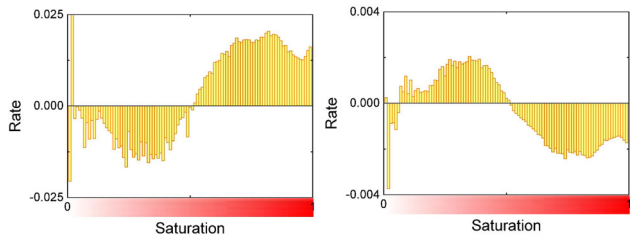
In the *Amazon* dataset, we remove the record before 2010. Time is discretized by weeks, and there are 237 time intervals in total. To validate the scalability of the model and give a comprehensive assessment, we split the dataset into several subsets by gender and categories of products (*Jewelry* dataset includes both jewelries and watches). Statistics of the datasets are shown in Table 2.

We then randomly split each dataset into training (80%), validation (10%), and test (10%) sets, and remove the cold items and users (items and users without records in training set) from the validation and test sets. The validation set is used for tuning hyper-parameters and the final performance comparison is conducted on the test set. The $F_1$-score and the normalized discounted cumulative gain (NDCG) are used to evaluate the performance of the baselines and our model. We recommend the top-$n$ items to each user to calculate $F_1$-score and NDCG for this user, and calculate the average score as the model performance. Our experiments are conducted by predicting Top-5, 10, 20, 50, and 100 favorite clothing.

### 6.2 Influential factors of aesthetics (RQ1)

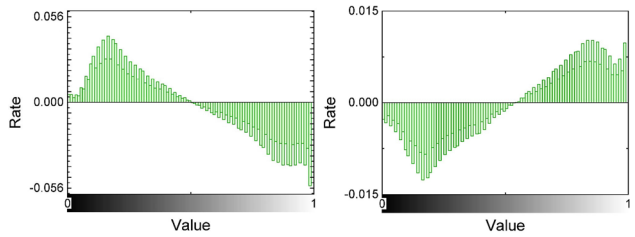
In this subsection, we explore what factors impact the users' aesthetics by reporting some statistics of the low-level aesthetics features: Hue, Saturation, and Value (HSV). Here, we use HSV rather than the high-level aesthetics features we extract because that the high-level features (high-dimensional vectors) are difficult to count and to represent, moreover, the specific meaning of each dimension is not clear. HSV is low-level yet representative, and makes the experiment result explainable.

Figure 6 shows the distribution of hue, saturation, and value, which are counted from the whole *Amazon* dataset (the *clothing shoes and jewelry* category). We normalize hue, saturation, and value into [0, 1] and normalize the histograms into a unit vector. The bar in the bottom of Fig. 6a is the hue, and different hue indicates different colors. From the figure, we can see that users prefer red and blue. The bar in the bottom of Fig. 6b is the saturation, which defines the brilliance and intensity of a color. From Fig. 6b, we can see that users prefer a lower saturation. The bar in the bottom of Fig. 6c is the value, which refers to the lightness or darkness of a color. The larger the value is, the lighter the color is. To present the difference of aesthetic preferences with certain



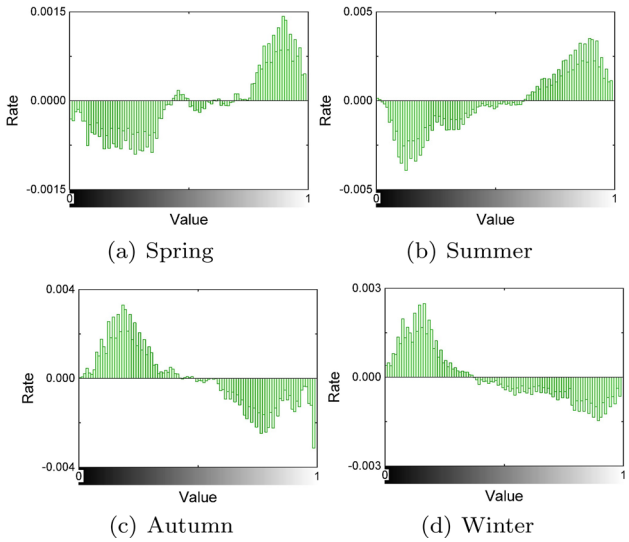

(a) Kids (b) Adults

**Fig. 7** Aesthetic preferences of users with different ages




(a) Men (b) Women

**Fig. 8** Aesthetic preferences of users with different genders




(a) Spring (b) Summer (c) Autumn (d) Winter

**Fig. 9** Aesthetic preferences in different seasons

factor, we report the difference between the normalized HSV histograms before and after the influence of certain factor, so there are positive values and negative values (see Figs. 7, 8, 9, 10). We mainly discuss the variation of HSV with different kinds of users and in different time.

#### 6.2.1 Influence of users

Modern recommender systems aim to provide the personalized recommendation, so the influence of different kinds of users is very important. It is obvious that different users have different aesthetic preferences. In this subsection, we show the variation of HSV impacted with the gender and age.

**Users with different ages:** Figure 7 shows the impact of users with different ages. Figure 7a, b shows the saturation distribution of kids and adults, respectively. Kids like clothes with really high saturation while adults like those with low saturation.

**Users with different genders:** Figure 8 presents the aesthetic preferences of males and females. Figure 8a shows the distribution of the value with males. They prefer dark clothes that can make them look mature and steady. Figure 8b shows the distribution with females. They prefer lovely and active clothes in light colors.

#### 6.2.2 Influence of time

For many products, especially clothes, movies, electronic devices, etc., sales change dramatically with time. Users' aesthetic preferences also change with time. For example, people like different colors and design in different seasons. Also, the fashion changes every year. In this subsection, we represent how time influences aesthetic preferences in a short term and long term.

**Seasonality:** Figure 9 represents users' aesthetic preferences in different seasons. Figure 9a–d shows the distribution of value in spring, summer, autumn, and winter, respectively. Users prefer light colors in spring and summer while prefer dark colors in autumn and winter.

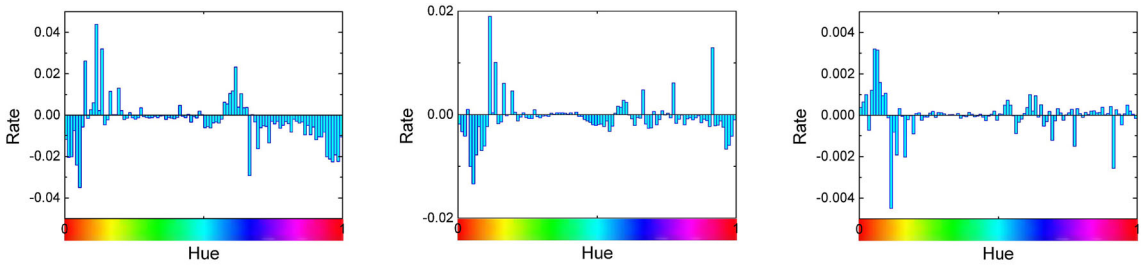
**Annual trend:** The fashion trend in different years is shown in Fig. 10. Histograms in Fig. 10a–c show the hue distribution of clothes in 2010, 2012, and 2014, respectively. As shown in Fig. 10, users preferred yellow and blue in 2010. In 2012, yellow and purple became popular. In 2014, the most popular color was red.

From the figures above, we come to the conclusion that users' aesthetic preferences change with different people and different time. So we propose a time-aware model taking these two factors into account as the basic model.

### 6.3 Performance of our model (RQ2)

To demonstrate the effectiveness of our model, we adopt the following methods as baselines for performance comparison:

- **BPR:** This **B**ayesian **P**ersonalized **R**anking method is a well-known ranking-based method [36] for implicit feedback. The preference pairs are constructed between the positive samples and the other ones. In our experiments, we optimize matrix factorization (MF) model with the pairwise optimization.
- **VBPR:** This **V**isual **B**ayesian **P**ersonalized **R**anking method is a visually aware recommendation method [14].




(a) 2010 (b) 2012 (c) 2014

**Fig. 10** Aesthetic preferences in different years

The visual features are pre-generated from the product images using CNN.

- **VNPR:** This **V**isual **N**eural **P**ersonalized **R**anking method is a visually aware neural network for recommendation [31]. We predict the user preference with both embeddings and visual features. Interactions of users and items are achieved by a deep structure.
- **DVBPR:** This **D**eep **V**isual **B**ayesian **P**ersonalized **R**anking method is an end-to-end visually aware neural recommendation model [19]. In [19], the embedding layer is removed and CNN is trained from the scratch to predict the user preference. Our experiments show that this setting is suboptimal, thus we reserve the embedding layer and pretrain the CNN on *ImageNet*.
- **CPLR:** This **C**ollaborative **P**airwise **L**earning to **R**ank method [23] is an extension of BPR. Collaborative information is used to improve the quality of negative sampling and further improve the quality of ranking.
- **WBPR:** **W**eighted **B**ayesian **P**ersonalized **R**anking [12] is an extension of BPR. WBPR improves the quality of negative sampling depending on the item popularity. Considering that popular items are unlikely to be neglected, WBPR gives larger confidence weights to negative samples with higher popularity.

For fair comparison, all models are tuned with the same strategy. We iterate 200 times to train all models. In each iteration, we enumerate all training data on the training set to update the model and select 1000 samples randomly from the test/validation set to test all models. We record the best performance of each model during this procedure as the evaluation of it. The sampling rate $\rho$ is set as 5 in our experiments. We show the $F_1$-score and NDCG with different $n$ in Figs. 11 and 12, respectively. Subfigures (a–f) show the performance on *Amazon*, *Men*, *Women*, *Clothes*, *Shoes*, and *Jewelry*, respectively. For all datasets and all models, we repeat our experiments 10 times. The bars in Figs. 11 and 12 indicate the average performance and the vertical lines on the top of the bars indicate the standard deviation. We can see that the datasets with higher sparsity show lower performance.

As we can see, BPR performs the worst since it only models user preference based on embeddings. Compared with BPR, advanced baselines including VBPR, VNPR, DVBPR, CPLR, and WBPR use extra information, or advanced sampling strategy, thus outperform it. VBPR, VNPR, DVBPR utilize visual features to model user visual preference and give prediction based on both visual features and embeddings. VBPR is the basic visual recommendation model, which simply injects visual features to an MF model. VNPR and DVBPR are deep visually aware models. VNPR inputs user and item embeddings into a deep neural network to achieve better interaction. However, VNPR fails to outperform VBPR in our experiments. DVBPR trains CNN in an end-to-end way to extract fashion-aware visual features. As shown in Figs. 11 and 12, DVBPR outperforms VBPR marginally.

CPLR and WBPR achieve better negative sampling, thus outperform BPR. CPLR utilizes the collaborative information to uncover the potential positive samples, and achieves significant improvement. WBPR weights samples based on item popularity and also performs better than BPR. However, the improvement is marginal since the strategy is simple and rough.

Enhanced by aesthetic features in both modeling and negative sampling aspects, our proposed VRA outperforms all baselines on all datasets. Taking *Jewelry* as example, the proposed VRA model outperforms the best baseline DVBPR about 7.16% on $F_1$-score@10 and 8.64% on NDCG@10.

An interesting observation is that we gain more improvement by the aesthetic feature on *Shoes* and *Clothes* subsets than on *Jewelry* subset. The possible reason is that compared with *Shoes* and *Clothes*, the style of *Jewelry* is relatively simple. For example, the color is almost either silver or golden. In this case, we gain less improvement by modeling the aesthetic features.

The sensitivity of $\lambda_c$ and $\lambda_r$ is shown in Fig. 13. To save space, we only show the result on *Jewelry* set. $\lambda_c$ is a weighting parameter for the coupled matrices (Fig. 13a). When $\lambda_c = 0.01$, our model achieves the best performance. The sensitivity of regularization coefficient $\lambda_r$ is shown in

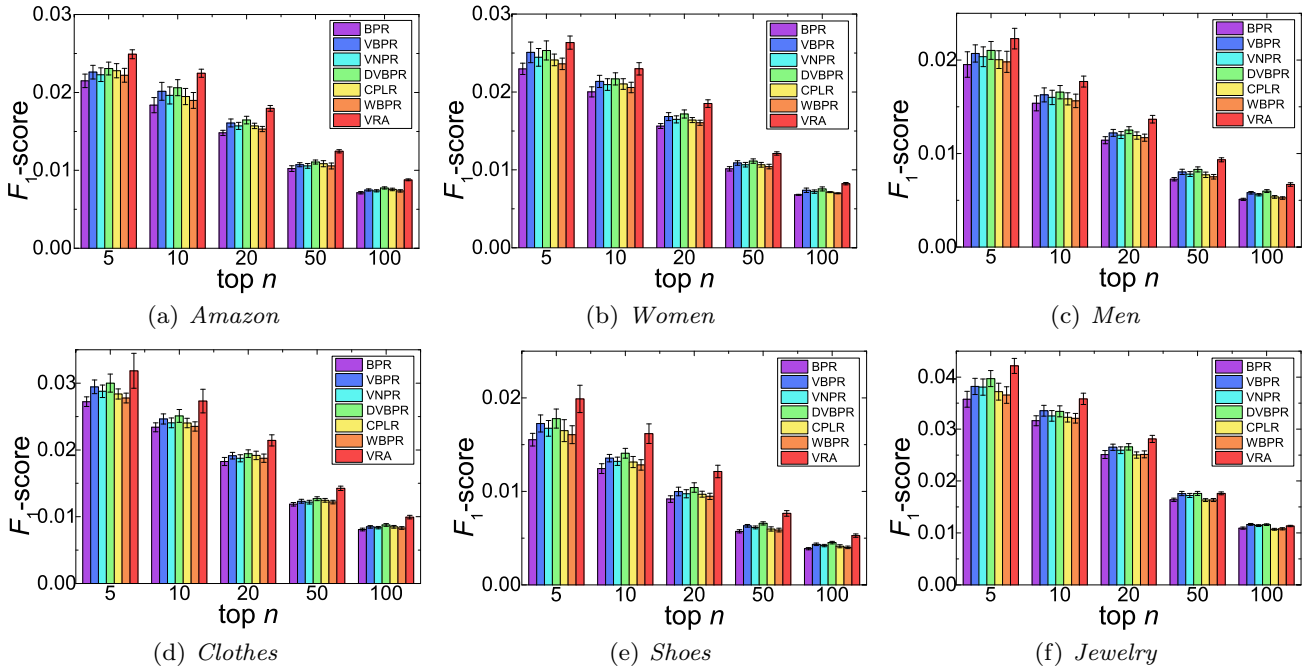



**Fig. 11** $F_1$-score of different datasets (test set)

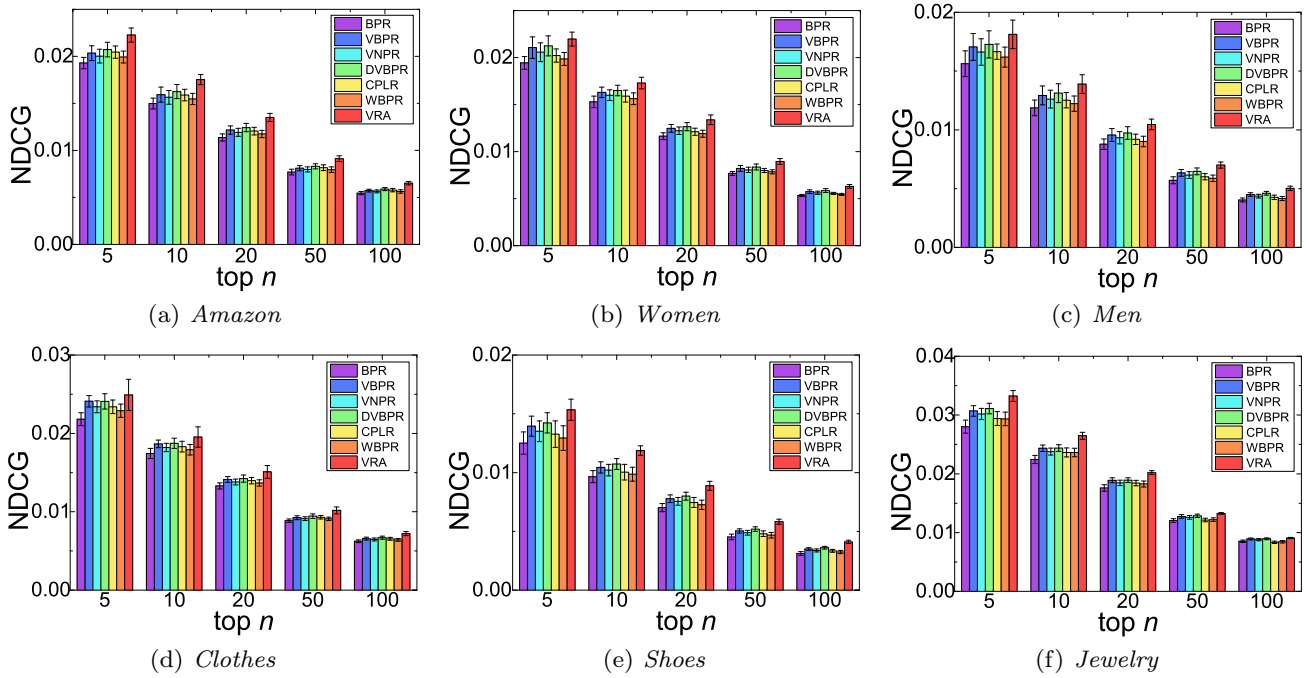
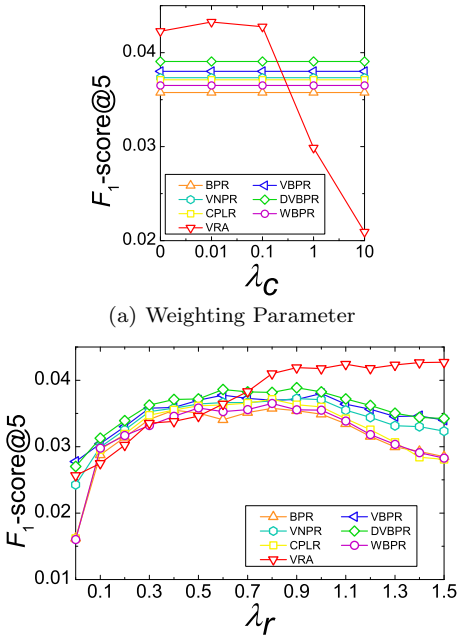



**Fig. 12** NDCG of different datasets (test set)

Fig. 13b. Our VRA achieves the best performance when $\lambda_r = 1.5$ and baselines achieve the best performance when $\lambda_r$ is around 0.9.

Figure 14 shows the performance with different lengths of latent factors. $K_1$ is the length of latent factors connecting users and items, $K_2$ is the length of latent factors connecting items and time. As Fig. 14 shows, the performance varies with $K_1$ obviously, while not so obviously with $K_2$. It may be because the rank of the user–item matrix $\mathbf{B}$ is much higher than that of the time–item matrix $\mathbf{C}$, and we need more representation ability to model users' preferences, so our model is more sensitive with $K_1$.

We can see the rank of *Jewelry* is small: when $K_1 = 50$, $K_2 = 20$, the model performs very well and with the increas-




(a) Weighting Parameter

(b) Regularization Coefficient

**Fig. 13** Impacts of hyper-parameters (*Jewelry*, validation set)

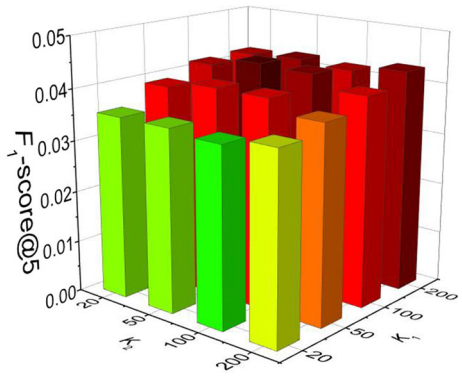



**Fig. 14** Performance with different lengths of latent factors (*Jewelry*, validation set)

ing of $K_1$ and $K_2$, the performance keeps stable. Considering the ranks of other datasets are higher (such as *All* and *Clothes*) thus larger $K_1$ and $K_2$ are required, we set $K_1 = 200$ and $K_2 = 200$ for all datasets.

## 6.4 Necessity of the aesthetic features (RQ3)

In this subsection, we discuss the necessity of the aesthetic features. We combine various widely used features to our basic model and compare the effectiveness of each feature by constructing five models:

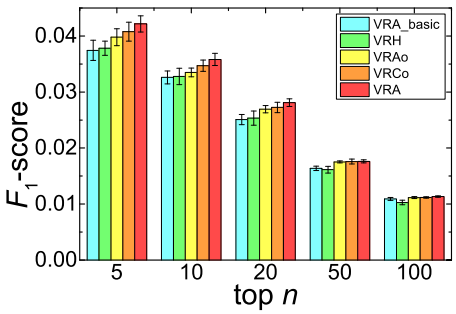



**Fig. 15** Performance of various features (*Jewelry*, test set)

- **VRA_basic:** This is our basic **V**isually Aware **R**ecommendation model without any visual features, which is represented in Sect. 4.1.
- **VRH:** This is a **V**isually Aware **R**ecommendation with Color **H**istograms. We only inject color histograms to our proposed basic model VRA_basic.
- **VRCo:** This is a **V**isually Aware **R**ecommendation with **C**NN Features **o**nly. We only inject CNN features to VRA_basic.
- **VRAo:** This is a **V**isually Aware **R**ecommendation with **A**esthetics Features **o**nly. We only inject aesthetic features to VRA_basic.
- **VRA:** This is our proposed model, utilizing both CNN features and aesthetic features.

Performances on *Jewelry* dataset are reported in Fig. 15. As we can see, VRA_basic performs the worst since no visual features are involved to provide the extra information. With the information of color distribution, VRH performs better, though still worse than VRCo and VRAo, because the low-level features are too crude and unilateral, and can provide very limited information about users' aesthetic preferences. VRCo and VRAo show the similar performance because both CNN features and aesthetic features have strong ability to mine user's preferences. Our VRA model, capturing both semantic information and aesthetic information, performs the best on the dataset.

Figure 15 shows that semantic and aesthetic information mutually enhance each other to a certain extent: they do not perform the best separately, yet they mutually enhance each other and achieve improvement together. Give an intuitive example, if a user wants to purchase a skirt, she needs to tell whether there is a skirt in the image (semantic information) when looking through products, and then she needs to evaluate if the skirt is good-looking and fits her tastes (aesthetic information) to make the final decision.

Several purchased and recommended items on *Amazon* dataset are represented in Fig. 16. The items in the first row are purchased by certain user (training data, the number is random). To illustrate the effectiveness of the aesthetic fea-


**Fig. 16** Items purchased by users and recommended by different models (*Amazon* dataset)

tures intuitively, we choose the users with explicit style of preferences and single category of items. The items in the second row and third row are recommended by VRCo and VRA, respectively. For these two rows, we choose the five best items from the 50 recommendations to exhibit. Comparing the first and the second row, we can see that leveraging semantic information, VRCo can recommend the congeneric (with the CNN features) and relevant (with tensor factorization) commodities. Although it can recommend the pertinent products, they are usually not in the same style with what the user has purchased. Capturing both aesthetic and semantic information, VRA performs much better. We can see that the items in the third row have more similar style with the training samples than the items in the second row.

Taking Fig. 16c as an example, we can see that the user prefers boots, ankle boots, or thigh boots. However, products recommended by VRCo are some different types of women's shoes, like high heels, snow boots, thigh boots, and cotton slippers. Though there is a thigh boot, it is not in line with the user's aesthetics due to the gaudy patterns and stumpy proportion, which rarely appears in her choices. The products recommended by VRA are better. First, almost all recommendations are boots. Then, thigh boots in the third row are in the same style with the training samples, like leather tex-

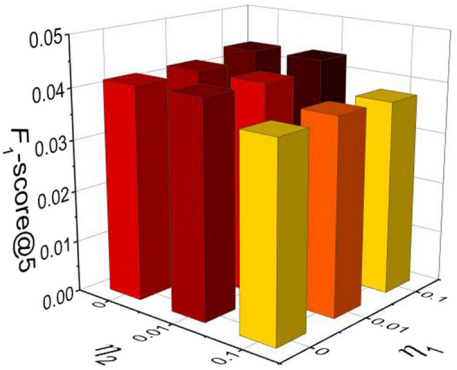



**Fig. 17** Influence of weighting parameters $\eta_1$ and $\eta_2$ (*Jewelry*, validation set)

ture, slender proportions, simple design, and some design elements of detail like straps and buckles (the second and third ones). Though the last one seems a bit different with the training samples, it is in the uniform style with them intuitively, since they are all designed for young ladies. It is also obvious in Fig. 16f, we can see that what the user likes are vibrant watches for young men. However, watches in the second row are in pretty different styles, like digital watches for children, luxuriantly decorated ones for ladies, old-fashioned ones for adults. Evidently, watches in the third row are in similar style with the train samples. They have similar color schemes and design elements, like the intricately designed dials, nonmetallic watchbands, small dials, and tachymeters. As we can see, with the aesthetic features and the CNN features complementing each other, VRA performs much better than VRCo.

### 6.5 Performance of APLR (RQ4)

In this subsection, we illustrate the effectiveness of our APLR optimization criterion.

Figure 17 shows the model tuning with respect to weighting parameters $\eta_1$ and $\eta_2$. We can see that when $\eta_1 = 0.1$ and $\eta_2 = 0.01$, the model achieves the best performance. When $\eta_1 = 0$ and $\eta_2 = 0$, the model becomes VRA_PLR. As shown in the figure, VRA_APLR outperforms VRA_PLR about 4.70% on $F_1$-score.

When $\eta_2$ is fixed, $F_1$-score usually takes the maximum when $\eta_1$ is about 0.1. When $\eta_1$ is fixed, $F_1$-score usually takes the maximum when $\eta_2$ is about 0.01. We come to the conclusion that the preference relationship $(p, q, r) \succ (p, \mathcal{N}_q, r)$ is more important than $(p, \mathcal{N}_q, r) \succ (p, \mathcal{Q}_{pr}^{-} \setminus \mathcal{N}_q, r)$.

## 7 Conclusion

In this paper, we investigated the usefulness of aesthetic features for personalized recommendation on implicit feedback datasets. We proposed a novel model that incorporates aesthetic features into a tensor factorization model to capture the aesthetic preferences of users at a particular time, and leveraged visual information and collaborative information to optimize the model. Experiments on challenging real-world datasets show that our proposed method dramatically outperforms state-of-the-art models, and succeeds in recommending items that fit users' style.

For the future work, we are interested in constructing high-order connections among items with spectrum clustering, social networks, etc. instead of only one-order connections, to enhance the pairwise learning. Also, we will establish a large dataset for product aesthetic assessment, and train the networks to extract the aesthetic information better. Lastly, we will investigate the effectiveness of the aesthetic features in the setting of explicit feedback.

**Acknowledgements** This work is supported in part by Beijing Outstanding Young Scientist Program NO. BJJWZYJH0 12019100020098 and National Natural Science Foundation of China (No. 61832017).

## References

1. Acar, E., Kolda, T.G., Dunlavy, D.M., Morup, M.: Scalable tensor factorizations for incomplete data. In: Chemometrics and Intelligent Laboratory Systems, pp. 41–56 (2010)
2. Acar, E., Kolda, T.G., Dunlavy, D.M.: All-at-once optimization for coupled matrix and tensor factorizations. In: Computing Research Repository—CORR (2011)
3. Bhargava, P., Phan, T., Zhou, J., Lee, J.: Who, what, when, and where: multi-dimensional collaborative recommendations using tensor factorization on sparse user-generated data. In: WWW, pp. 130–140 (2015)
4. Buchanan, A.M., Fitzgibbon, A.W.: Damped newton algorithms for matrix factorization with missing data. CVPR **2**, 316–322 (2005)
5. Cao, Z., Qin, T., Liu, T.Y., Tsai, M.F., Li, H.: Learning to rank: from pairwise approach to listwise approach. In: ICML, pp. 129–136 (2007)
6. Chen, J., Zhang, H., He, X., Nie, L., Liu, W., Chua, T.S.: Attentive collaborative filtering: multimedia recommendation with item- and component-level attention. In: SIGIR, pp. 335–344 (2017)
7. Chen, T., He, X., Kan, M.Y.: Context-aware image tweet modelling and recommendation. In: MM, pp. 1018–1027 (2016)
8. Chen, X., Wang, P., Qin, Z., Zhang, Y.: Hlbpr: A hybrid local Bayesian personal ranking method. In: WWW, pp. 21–22 (2016)
9. Chen, X., Zhang, Y., Ai, Q., Xu, H., Yan, J., Qin, Z.: Personalized key frame recommendation. In: SIGIR, pp. 315–324 (2017)
10. Datta, R., Joshi, D., Li, J., Wang, J.Z.: Studying aesthetics in photographic images using a computational approach. In: ECCV, pp. 288–301 (2006)
11. Ding, J., Feng, F., He, X., Yu, G., Li, Y., Jin, D.: An improved sampler for Bayesian personalized ranking by leveraging view data. In: WWW, pp. 13–14 (2018)

12. Gantner, Z., Drumond, L., Freudenthaler, C., Schmidt-Thieme, L.: Bayesian personalized ranking for non-uniformly sampled items. In: KDDCup (2011)
13. Goldberg, D.: Using collaborative filtering to weave an information tapestry. In: Communications of the ACM, pp. 61–70 (1992)
14. He, R., McAuley, J.: VBPR: visual Bayesian personalized ranking from implicit feedback. In: AAAI, pp. 144–150 (2016)
15. He, X., Zhang, H., Kan, M.Y., Chua, T.S.: Fast matrix factorization for online recommendation with implicit feedback. In: SIGIR, pp. 549–558 (2016)
16. He, X., Liao, L., Zhang, H., Nie, L., Hu, X., Chua, T.S.: Neural collaborative filtering. In: WWW, pp. 173–182 (2017)
17. He, X., He, Z., Song, J., Liu, Z., Jiang, Y.G., Chua, T.S.: Nais: neural attentive item similarity model for recommendation. IEEE Trans. Knowl. Data Eng. **66**, 1 (2018)
18. Jagadeesh, V., Piramuthu, R., Bhardwaj, A., Di, W., Sundaresan, N.: Large scale visual recommendations from street fashion images. In: SIGKDD, pp. 1925–1934 (2014)
19. Kang, W., Fang, C., Wang, Z., McAuley, J.: Visually-aware fashion recommendation and design with generative image models. In: ICDM, pp. 207–216 (2017)
20. Karatzoglou, A., Amatriain, X., Baltrunas, L., Oliver, N.: Multiverse recommendation: N-dimensional tensor factorization for context-aware collaborative filtering. In: RecSys, pp. 79–86 (2010)
21. Kolda, T.G., Bader, B.W.: Tensor decompositions and applications. SIAM Rev. **6**, 455–500 (2009)
22. Koren, Y.: The Bellkor solution to the Netflix grand prize. In: Netflix Prize Documentation, pp. 1–10 (2009)
23. Liu, H., Wu, Z., Zhang, X.: Cplr: collaborative pairwise learning to rank for personalized recommendation. In: Knowledge-Based Systems (2018)
24. Liu, J., Wu, C., Xiong, Y., Liu, W.: List-wise probabilistic matrix factorization for recommendation. Inf. Sci. **6**, 434–447 (2014)
25. Lu, X., Lin, Z., Jin, H., Yang, J., Wang, J.Z.: RAPID: rating pictorial aesthetics using deep learning. In: MM, pp. 457–466 (2014)
26. Luo, W., Wang, X., Tang, X.: Content-based photo quality assessment. IEEE Trans. Multimed. **45**, 1930–1943 (2013)
27. Ma, S., Liu, J., Chen, C.W.: A-lamp: Adaptive layout-aware multi-patch deep convolutional neural network for photo aesthetic assessment. In: CVPR, pp. 722–731 (2017)
28. Marchesotti, L., Perronnin, F., Larlus, D., Csurka, G.: Assessing the aesthetic quality of photographs using generic image descriptors. In: ICCV, pp. 1784–1791 (2011)
29. McAuley, J., Targett, C., Shi, Q., van den Hengel, A.: Image-based recommendations on styles and substitutes. In: SIGIR, pp. 43–52 (2015)
30. Murray, N., Marchesotti, L., Perronnin, F.: Ava: A large-scale database for aesthetic visual analysis. In: CVPR, pp. 2408–2415 (2012)
31. Niu, W., Caverlee, J., Lu, H.: Neural personalized ranking for image recommendation. In: WSDM, pp. 423–431 (2018)
32. Pan, W., Chen, L.: Gbpr: group preference based Bayesian personalized ranking for one-class collaborative filtering. In: IJCAI, pp. 2691–2697 (2013)
33. Pan, W., Zhong, H., Xu, C., Ming, Z.: Adaptive Bayesian personalized ranking for heterogeneous implicit feedbacks. Knowl. Based Syst. **35**, 173–180 (2015)
34. Qiu, S., Cheng, J., Yuan, T., Leng, C., Lu, H.: Item group based pairwise preference learning for personalized ranking. In: SIGIR, pp. 1219–1222 (2014)
35. Rendle, S., Schmidt-Thieme, L.: Pairwise interaction tensor factorization for personalized tag recommendation. In: WSDM, pp. 81–90 (2010)
36. Rendle, S., Freudenthaler, C., Gantner, Z., Schmidt-Thieme, L.: BPR: bayesian personalized ranking from implicit feedback. In: UAI, pp. 452–461 (2009)
37. Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Item-based collaborative filtering recommendation algorithms. In: WWW, pp. 285–295 (2001)
38. Sha, D., Wang, D., Zhou, X., Feng, S., Zhang, Y., Yu, G.: An approach for clothing recommendation based on multiple image attributes. In: WAIM, pp. 272–285 (2016)
39. Wang, Z., Chang, S., Dolcos, F., Beck, D., Liu, D., Huang, T.S.: Brain-inspired deep networks for image aesthetics assessment. Mich. Law Rev. **23**, 123–128 (2016)
40. Xiong, L., Chen, X., Huang, T.K., Schneider, J.G., Carbonell, J.G.: Temporal collaborative filtering with bayesian probabilistic tensor factorization. In: SDM, pp. 211–222 (2010)
41. Yu, W., Qin, Z.: Graph convolutional network for recommendation with low-pass collaborative filters. In: ICML, pp. 10936–10945 (2020a)
42. Yu, W., Qin, Z.: Sampler design for implicit feedback data by noisy-label robust learning. In: SIGIR, pp. 861–870 (2020b)
43. Yu, W., Zhang, H., He, X., Chen, X., Xiong, L., Qin, Z.: Aesthetic-based clothing recommendation. In: WWW, pp. 649–658 (2018)
44. Yu, W., Lin, X., Ge, J., Ou, W., Qin, Z.: Semi-supervised collaborative filtering by text-enhanced domain adaptation. In: KDD, pp. 2136–2144 (2020)
45. Zhang, W., Chen, T., Wang, J., Yu, Y.: Optimizing top-n collaborative filtering via dynamic negative item sampling. In: SIGIR, pp. 785–788 (2013)
46. Zhao, L., Lu, Z., Pan, S.J., Yang, Q.: Matrix factorization+ for movie recommendation. In: IJCAI, pp. 3945–3951 (2016)
47. Zhao, T., McAuley, J., King, I.: Leveraging social connections to improve personalized ranking for collaborative filtering. In: CIKM, pp. 261–270 (2014)