World Wide Web (2021) 24:1885–1901
https://doi.org/10.1007/s11280-021-00913-3

Attribute-aware explainable complementary clothing recommendation

Yang Li[1] · **Tong Chen**[1] · **Zi Huang**[1]

Received: 30 October 2020 / Revised: 3 March 2021 / Accepted: 10 June 2021 /
Published online: 28 July 2021

Abstract

Modelling mix-and-match relationships among fashion items has become increasingly demanding yet challenging for modern E-commerce recommender systems. When performing clothes matching, most existing approaches leverage the latent visual features extracted from fashion item images for compatibility modelling, which lacks explainability of generated matching results and can hardly convince users of the recommendations. Though recent methods start to incorporate pre-defined attribute information (e.g., colour, style, length, etc.) for learning item representations and improving the model interpretability, their utilisation of attribute information is still mainly reserved for enhancing the learned item representations and generating explanations via post-processing. As a result, this creates a severe bottleneck when we are trying to advance the recommendation accuracy and generating fine-grained explanations since the explicit attributes have only loose connections to the actual recommendation process. This work aims to tackle the explainability challenge in fashion recommendation tasks by proposing a novel Attribute-aware Fashion Recommender (AFRec). Specifically, AFRec recommender assesses the outfit compatibility by explicitly leveraging the extracted attribute-level representations from each item's visual feature. The attributes serve as the bridge between two fashion items, where we quantify the affinity of a pair of items through the learned compatibility between their attributes. Extensive experiments have demonstrated that, by making full use of the explicit attributes in the recommendation process, AFRec is able to achieve state-of-the-art recommendation accuracy and generate intuitive explanations at the same time.

Keywords Clothing recommendation · Explainable recommender systems

This article belongs to the Topical Collection: *Special Issue on Explainability in the Web*
Guest Editors: Guandong Xu, Hongzhi Yin, Irwin King, and Lin Li

✉ Tong Chen
tong.chen@uq.edu.au

Yang Li
yang.li@uq.edu.au

Zi Huang
huang@itee.uq.edu.au

[1] The University of Queensland, Brisbane, Australia

1 Introduction

The advancement of modernisation attracts a rapidly growing attention to fashion. A wide range of fashion-focused social websites have emerged in recent decades, such as Polyvore [1] and ShopLook[2]. With an overwhelming amount of product choices, customers nowadays are craving for personal advice on outfit matching and recommendation of the most suitable item for their wardrobes, which brings in a great opportunity of designing automated tools for measuring fashion compatibility.

Recent research in the fashion area evolves from fundamental clothing recognition [15, 28], style understanding [2] to aesthetic and compatibility analysis [5, 11, 21, 22]. Learning compatibility relationships is a challenging and sophisticated task, as whether two clothes (e.g., top and bottom clothes) are a good match is usually determined by a complex mixture of various factors. A large body of work on this task models compatibility notions by computing latent representations for a given pair of items, then modelling the similarity between items via those representations [5, 11, 22]. In this regard, latent factor models, especially deep models [5, 20] have commonly demonstrated promising recommendation accuracy. However, a main drawback of these latent factor methods is that the recommendation process is non-transparent to users, making it hard for users to justify the reasons behind successfully matched clothes. In the real-world scenario, users usually not only want to know whether two outfits are compatible or not but also would like to understand the major factors that lead to the failure or success of matching.

Though visual explanations (usually made with attention) are offered in some recent methods to reveal a model's inner mechanism and perform model validation [9], however, they are less helpful for convincing users of the generated clothes matching results and making detailed explanations beyond only the appearance of items. In fact, as illustrated in Figure 1, the property of a fashion item can be further decomposed into multiple fine-grained attributes (e.g., shape, colour, pattern, material, etc.), which are highly relevant when users are shopping for clothes. To enhance the model interpretability, some work attempts to incorporate information of pre-defined attributes of clothes when modelling clothes compatibility. However, despite the availability of attribute information, the attributes are only involved in the recommendation process in the form of latent features of items, thus giving up the rich compatibility signals between explicit attributes and making the generated explanation coarse-grained. For example, [4] generates explanations by post-processing the associated attributes after a recommendation is made, making the attribute-wise explanations loosely connected to the actual recommendation results. Meanwhile, [26] requires pretraining an individual decision tree before meaningful attribute combinations can be used for clothes matching and interpretation, and the quality of both recommendation and explanation is highly dependent on the selected decision tree model.

To alleviate the aforementioned limitations of previous work, we introduce our Attribute-aware Fashion Recommender (AFRec), which makes full use of explicit attribute information to mimic a human's decision-making process where the compatibility of two clothes are usually determined by comparing various attributes of both items. Specifically, taking the images of a pair of clothes as the input, AFRec utilises a pretrained convolutional neural network (CNN) to extract visual features from both clothes. Then, we design an innovative

[1] https://www.polyvore.com
[2] https://www.shoplook.io

Figure 1 An example of clothing attribute

semantic attribute extractor that automatically maps each item to a group of attribute representations. Unlike existing attribute-based methods that directly fuse extracted attributes into a unified representation for each item [4, 26], we disentangle the straightforward item-item affinity into the explicit attribute-attribute compatibility. To achieve this, we propose a novel attribute-wise reciprocal attention module, where the affinity between two items is conditioned on the inherent compatibility of each attribute pair as well as each item's performance across all attributes. This enables AFRec to precisely bridge two complementary clothes with fine-grained attributes. Moreover, the pairwise attribute compatibility scores allow AFRec to provide intuitive attribute-level explanations on the recommendation results.

Our main contributions are summarised as follows:

- We approach an emerging and important research problem - explainable complementary clothes recommendation from a different view, i.e., using attribute-level compatibility to bridge two complementary clothes.
- We propose Attribute-aware Fashion Recommender (AFRec), a novel model that explores the fine-grained attribute-level collocation via a CNN-based semantic attribute extractor, which is followed by an innovative attribute-wise attentive compatibility modelling paradigm for clothes matching.
- We extensively evaluate AFRec on two benchmark datasets, where the results suggest that it is able to outperform state-of-the-art baselines and generate intuitive explanations at the same time.

2 Related work

Existing work on recommending complementary clothing items mainly utilises the visual signals extracted from the product image data to model the visual correlations between items and user preferences. McAuley et al. [16] propose to use Low-rank Mahalanobis Transformation to learn a latent style space for minimising the distance between matched clothing item embeddings and maximising that of mismatched ones. Veit et al. [25] employ the Siamese CNNs to learn a metric for compatibility measurement in an end-to-end manner. Some researchers argue that the complex compatibility relationships cannot be captured by directly learning a single latent space. He et. al [7] propose to learn a mixture of multiple metrics with weight confidences to model the relationships between heterogeneous

items. Veit et al. [24] propose Conditional Similarity Network, which learns disentangled item features whose dimensions can be used for separate similarity measurements. Li et al. [11] use an encoder to fuse features from multimodal inputs and adopt pooling techniques to get a single representation of an outfit for compatibility measurement. Vasileva et al. [23] claim that respecting type information has important consequences. Thus, they build type-wise trainable mask embeddings and use them to attend on different latent aspects when measuring different kinds of top-bottom pairs. Similarly, Yang et al. [27] introduce a translation-based type-aware model, which learns type-specific embeddings to connect compatible item embedding pairs. Different from the previous category-aware work [23, 27], instead of learning either mask or categorical relation embeddings, we build category-specific weight matrices in AFRec, which help the model to focus on different latent aspects for attribute representation pairs in different categorical groups.

However, there are some voices arguing that these previous methods suffer from limited interpretability. Han et al. [4] propose a Bayesian Personalised Ranking (BPR) framework named PAICM that adopts NMF to learn the latent attribute-level prototype embeddings for both compatible and incompatible outfits. Thus, the model could provide recommendation explanation by comparing the item-level embedding with the closest prototype embedding. However, since the interpretability of this method highly relies on the quality of the learned prototype embeddings, the model is sensitive to the number of defined prototypes. Xun et al. [26] propose to draw harmonious matching rules through a deep decision tree for the explainability of the recommendation model. Another explainable fashion recommendation model [13] learns to generate review comments by an attentive RNN-based decoder using the fused item-level embeddings. Nevertheless, these approaches either require abundant well-annotated attribute labels of each item for matching rule mining or user-generated reviews for training the explanation generation component. This impedes the practicality of those methods on most fashion datasets, where only a short textual description is available for each clothing item. Different from those methods, our model innovatively captures the fine-grained pairwise interactions at the attribute level, which provides an explicit and clear explanation by automatically concentrating on the most important attribute factors in a given compatible/incompatible outfit pair.

3 Problem formulation

In this paper, we focus on the widely studied problem of matching top and bottom clothes [1, 12, 14, 26], while our approach can be easily generalised to other types of clothes matching problems. Let us use $\mathcal{T} = \{t_1, t_2, ..., t_{N^t}\}$, $\mathcal{B} = \{b_1, b_2, ..., b_{N^b}\}$, $\mathcal{A} = \{a_1, a_2, ..., a_K\}$ and $\mathcal{C} = \{c_1, c_2, ..., c_{|\mathcal{C}|}\}$ to denote the set of top images, bottom images, attributes and item categories in the dataset, $\mathcal{D}$, where N^t, N^b, K and L are the total numbers of tops, bottoms, attributes and item categories, respectively. Bold lowercase letters and bold uppercase letters are used to indicate embedding vectors and matrices, respectively.

In this work, we target at modelling outfit compatibility as well as exploring the explainability of the generated recommendations. Formally, given an arbitrary top-bottom pair (t_i, b_j), our model is able to utilise the attribute information $\mathcal{A}$ associated with each item to distinguish whether t_i and b_j is a qualified match or not. In the case of the ranking task, our model is expected to generate the highest ranking score for a ground truth item pair than a non-matching item pair.

4 Proposed approach

As discussed in Section 1, most existing work models fashion compatibility by measuring the similarity between fashion items' latent representations, where the meaning of the features is incomprehensible to users. As a result, they could hardly provide convincing explanations for their predictions. To address this limitation, we propose an attribute-aware fashion recommender, namely AFRec, which supports comprehensive clothes matching and reasoning at the attribute level. The workflow of AFRec is shown in Figure 2. In this section, we first introduce the global and attribute-specific representation extraction procedure. Then, we describe our designed attribute reciprocal attention mechanism, which fully explores the complementary correlations between the top and bottom attributes for compatibility modelling. Finally, we give the learning objective for training our model.

4.1 Item visual feature extraction

As illustrated in the left part of Figure 2, we first utilise a pretrained CNN to extract high-level visual features from the raw input images. Considering both performance and computational complexity, we adopt ResNet-18 [6] pretrained on ImageNet dataset [19] as the backbone module. Accordingly, for image t_i/b_j that are of the size 224×224 with 3 colour channels, the feature maps output from the pretrained CNN can be represented as $\mathbf{F}_{t_i} \in \mathbb{R}^{D\times 7\times 7}$ and $\mathbf{F}_{b_j} \in \mathbb{R}^{D\times 7\times 7}$, where D is the output dimension size ($D = 512$ in a typical ResNet-18), and 7×7 denotes the output feature map size, i.e., height $\times$ width.

Generating global item embeddings To compress the visual feature maps into a compact item embedding, we use two sets $\mathcal{F}_{t_i}$ and $\mathcal{F}_{b_j}$ to collect all $7 \times 7 = 49$ D-dimensional feature vectors, i.e., $\mathcal{V}_{t_i} = \left\{\mathbf{v}_1^{t_i}, \mathbf{v}_2^{t_i}, ..., \mathbf{v}_{49}^{t_i}\right\}$ and $\mathcal{V}_{b_j} = \left\{\mathbf{v}_1^{b_j}, \mathbf{v}_2^{b_j}, ..., \mathbf{v}_{49}^{b_j}\right\}$ where $\mathbf{v} \in \mathbb{R}^D$ corresponds to one feature in the feature map. Then, the global embedding vectors of items

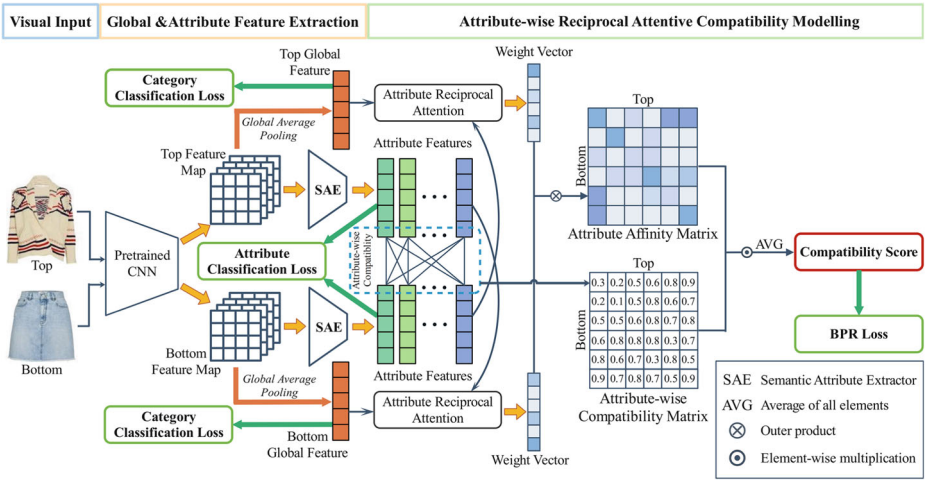

Figure 2 An overview of our proposed AFRec model

t_i and b_j can be obtained by feeding $\mathcal{V}_{t_i}$ and $\mathcal{V}_{b_j}$ into a global average pooling layer:

$$\mathbf{v}_{t_i} = \frac{1}{49}\sum_{n=1}^{49}\mathbf{v}_n^{t_i}, \quad \mathbf{v}_{b_j} = \frac{1}{49}\sum_{n=1}^{49}\mathbf{v}_n^{b_j}, \tag{1}$$

where $\mathbf{v}_{t_i}^{global}, \mathbf{v}_{b_j}^{global} \in \mathbb{R}^D$ denote the global feature embedding for t_i and b_j, respectively.

Fine-tuning pretrained CNN As the pretrained ResNet-18 is not originally designed for attribute-aware fashion recommendation, we fine-tune this CNN module with an item categorisation task. The rationale is that, fashion items of different categories tend to demonstrate different distributions over attributes. For instance, "sleeve length" is an important attribute for shirts and sweaters, while people tend to pay more attention to the "waistline" of a dress. This requires the model to focus on different attributes when handling different types of clothes. Therefore, to generate category-sensitive and more discriminative item embeddings to better guide the subsequent attribute extraction procedure, we design an additional item classification task with cross-entropy loss, which is used to fine-tune the pretrained CNN module:

$$\begin{aligned} \widehat{\mathbf{y}}_{item} &= \text{softmax}\left(\mathbf{W}^{cat}\mathbf{v}_{item}^{global} + \mathbf{b}^{cat}\right), \\ \mathcal{L}_{category} &= -\sum_{\forall item \in \mathcal{T}\cup\mathcal{B}} \mathbf{y}_{item}^{\top}\log(\widehat{\mathbf{y}}_{item}), \end{aligned} \tag{2}$$

where $\mathbf{W}^{cat} \in \mathbb{R}^{|\mathcal{C}|\times D}$ and $\mathbf{b}^{cat} \in \mathbb{R}^{|\mathcal{C}|}$ are the weight and bias of the classifier, $\widehat{\mathbf{y}}_{item} \in \mathbb{R}^{|\mathcal{C}|}$ is the predicted probability distribution over all item categories, and $\mathbf{y}_{item}$ is an one-hot encoding of each item's ground truth category label.

4.2 Semantic attribute representation extraction

On e-commerce websites, on top of visual information (i.e., images), a fashion garment usually has a textual description at the same time. This allows us to effectively summarise meaningful item attributes such as shape, pattern and style. With a pre-defined item attribute set $\mathcal{A}$, we propose a CNN-based semantic attribute extractor (SAE) for meaningful attribute-specific region localisation and representation generation in a weakly-supervised manner. Previous attribute-aware solutions [4, 26] learns universal representation for every single attribute, and use the combinatorial feature of different attributes for item representation learning. However, using fixed attribute representations lacks adequate flexibility as each item may exhibit different characteristics towards each attribute. Hence, in AFRec, we allow each item to have its unique representation regarding an attribute a_k, which is learned in an attribute-specific feature space.

As illustrated in Figure 3, the extracted feature map $\mathbf{F} \in \mathbb{R}^{D\times 7\times 7}$ is shared over all attribute-specific blocks (each block is marked by blue lines in Figure 3). There are K blocks defined in SAE corresponding to K fashion attributes. For the k-th attribute $a_k \in \mathcal{A}$, we adopt an independent convolutional layer whose kernel size is of $D \times 1 \times 1$ to transform the visual feature map $\mathbf{F}$ to $\mathbf{F}'_k \in \mathbb{R}^{D\times 7\times 7}$. Note that the convolutional layer in each attribute block has a unique set of parameters. Then, with a global average pooling operation as in (1), we can obtain attribute a_k's embedding vector $\mathbf{a}_k \in \mathbb{R}^D$. Similarly, the same attribute extraction scheme is applied in all other blocks. Accordingly, for both items t_i and b_j, we stack all K attribute representations obtained from SAE into two $K \times D$ matrices, i.e.,

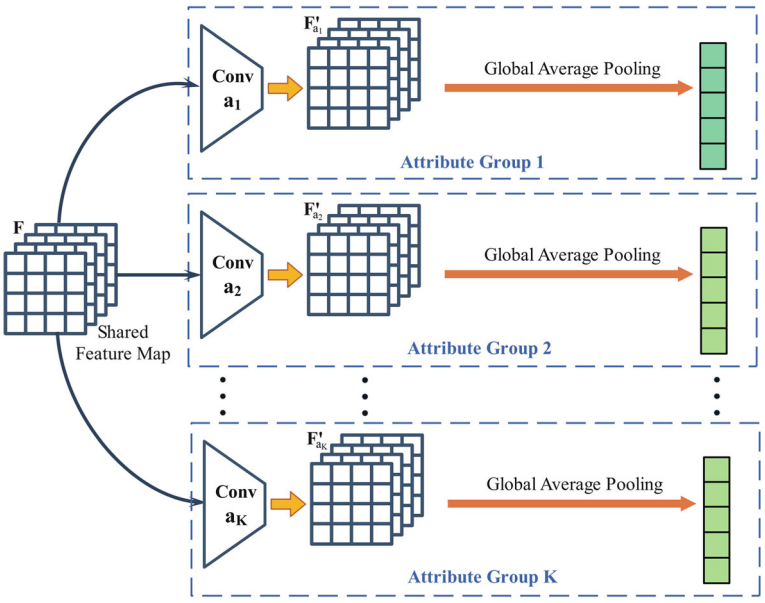

Figure 3 An overview of Semantic Attribute Extractor (SAE)

$\mathbf{A}_{t_i} = \left[\mathbf{a}_1^{t_i}, \mathbf{a}_2^{t_i}, ..., \mathbf{a}_K^{t_i}\right]$ and $\mathbf{A}_{b_j} = \left[\mathbf{a}_1^{b_j}, \mathbf{a}_2^{b_j}, ..., \mathbf{a}_K^{b_j}\right]$. Intuitively, $\mathbf{A}_{t_i}$ and $\mathbf{A}_{b_j}$ can be viewed as two attribute-aware feature matrices representing t_i and b_j.

Apparently, we can directly optimise each $\mathbf{a}_k$ within $\mathbf{A}_{t_i}$ and $\mathbf{A}_{b_j}$ using downstream clothes matching tasks. However, to ensure sufficient expressiveness of the learned attribute representation $\mathbf{a}_k$, we further introduce a prediction task in the SAE layer. To be specific, for each attribute, e.g., a_k ="colour", we can obtain the ground truth label (i.e., value) from the corresponding item, e.g., "colour"→"white". Suppose for the k-th attribute, there are N^k possible values, then we can use one-hot encoding $\mathbf{z}_k^{item}$ to label the observed value on $item \in \mathcal{T} \cup \mathcal{B}$. In a similar vein to Section 4.1, we perform attribute value prediction with cross-entropy loss:

$$\begin{aligned} \widehat{\mathbf{z}}_k^{item} &= \text{softmax}(\mathbf{W}_k^{attr}\mathbf{a}_k^{item} + \mathbf{b}_k^{attr}), \\ \mathcal{L}_{attribute} &= -\sum_{\forall item \in \mathcal{T} \cup \mathcal{B}} \sum_{k=1}^{K} \mathbf{z}_k^{item\top} \log(\widehat{\mathbf{z}}_k^{item}), \end{aligned} \tag{3}$$

where $\mathbf{W}_k^{attr} \in \mathbb{R}^{K \times D}$ and $b_k^{attr} \in \mathbb{R}^{N^k}$ are the weight and bias of the classifier for the k-th attribute, $\widehat{\mathbf{z}}_k^{item} \in \mathbb{R}^{N^k}$ denotes the item's estimated probability distribution over all possible values of attribute a_k. By optimising $\mathcal{L}_{attribute}$, we can effectively enforce the attribute embedding learned in each block to be a high-quality reflection on the k-th attribute of the target item.

4.3 Attribute-wise reciprocal attention

As a common practice, people tend to consider the different combinations of top and bottom attributes when choosing clothes to wear. For example, when a person wants to find a pair

of pants to match his/her T-shirt, he/she may think about whether the colour and the pattern of the pants are compatible with the T-shirt. Inspired by the recent advances of attention mechanism in computer vision [3, 8, 29], we propose an attribute-wise reciprocal attention mechanism for clothes matching. In particular, for top t_i's attribute representation $\mathbf{a}_k^{t_i} \in \mathbf{A}_{t_i}$, an attention score $s_k^{t_i}$ representing its importance to the bottom b_j can be computed via the following:

$$s_k^{t_i} = \mathbf{w}^\top \tanh\left(\mathbf{W}_1 \mathbf{a}_k^{t_i} + \mathbf{W}_2 \mathbf{v}_{b_j}^{global}\right), \tag{4}$$

where $\mathbf{w} \in \mathbb{R}^D$ carries the projection weight, while $\mathbf{W}_1 \in \mathbb{R}^{D\times D}$ and $\mathbf{W}_2 \in \mathbb{R}^{D\times D}$ are two weight matrices. Then, a normalised reciprocal attention weight $\alpha_k^{t_i}$ for t_i's attribute $a_k^{t_i}$ is calculated as follows:

$$\alpha_k^{t_i} = \frac{\exp\left(s_k^{t_i}\right)}{\sum_{k=1}^{K} \exp\left(s_k^{t_i}\right)}. \tag{5}$$

Now, we can get t_i's reciprocal attribute attention weight vector $\mathbf{v}_{t_i}^{attn} = \left[\alpha_1^{t_i}, \alpha_2^{t_i}, ..., \alpha_K^{t_i}\right] \in \mathbb{R}^K$, where the value of k-th dimension in $\mathbf{v}_{t_i}^{attn}$ represents t_i's performance on the k-th attribute regarding the bottom b_j. Similarly, we can also obtain b_j's attribute attention vector $\mathbf{v}_{b_j}^{attn}$:

$$\begin{aligned} s_k^{b_j} &= \mathbf{v}_{attn}^\top \tanh\left(\mathbf{W}_1 \mathbf{a}_k^{b_j} + \mathbf{W}_2 \mathbf{v}_{t_i}^{global}\right), \\ \alpha_k^{b_j} &= \frac{\exp\left(s_k^{b_j}\right)}{\sum_{k=1}^{K} \exp\left(s_k^{b_j}\right)}, \\ \mathbf{v}_{b_j}^{attn} &= \left[\alpha_1^{b_j}, \alpha_2^{b_j}, ..., \alpha_K^{b_j}\right] \in \mathbb{R}^K. \end{aligned} \tag{6}$$

So far, $\mathbf{v}_{t_i}^{attn}$ and $\mathbf{v}_{b_j}^{attn}$ can be viewed as attribute-aware representations of t_i and b_j, which are respectively conditioned on each other.

4.4 Explicit attribute-aware compatibility modelling

With the obtained attribute representations $\mathbf{A}_{t_i}$ and $\mathbf{A}_{b_j}$, we then perform the compatibility prediction in a pairwise manner, which is illustrated in the right part of Figure 2. To be specific, for each top-bottom pair (t_i, b_j), we first project their attribute representations into a category-specific space via linear transformation, then we obtain a compatibility matrix $\mathbf{M}_{ij}^{compat} \in \mathbb{R}^{K\times K}$ by calculating an affinity score between every pair of attribute-wise representations $\left(\mathbf{a}_k^{t_i}, \mathbf{a}_{k'}^{b_j}\right)$. To allow for efficient computation, the matrix-level computation is written as:

$$\mathbf{M}_{ij}^{compat} = (\mathbf{A}_{t_i}\mathbf{W}^{c_i c_j})\mathbf{W}^{compat}(\mathbf{A}_{b_j}\mathbf{W}^{c_i c_j})^\top \in \mathbb{R}^{K\times K}, \tag{7}$$

where $\mathbf{W}^{c_i c_j} \in \mathbb{R}^{D\times D}$ is a category-specific weight matrix, here, $c_i c_j$ denotes the pair of (t_i, b_j)'s categorical labels, $\mathbf{W}^{compat} \in \mathbb{R}^{D\times D}$ is a transformation matrix that aligns the feature spaces for both attribute-wise feature matrices for compatibility measurement. Each element $m_{kk'}^{compat} \in \mathbf{M}_{ij}^{compat}$ results from the dot product between the intrinsic attribute representations $\mathbf{a}_k^{t_i}$ and $\mathbf{a}_{k'}^{b_j}$. Hence, $m_{kk'}^{compat}$ can be viewed as the inherent compatibility score for attribute pair $(a_k, a_{k'})$, which is learned with the contexts given by the (t_i, b_j) pair. A higher score means that two attributes are closely correlated for clothes matching, e.g.,

attributes "bottom length" and "waistline" are usually bounded when matching the sizing of clothes.

Moreover, recall that we have also obtained the attention vectors $\mathbf{v}_{attn}^{t_i}, \mathbf{v}_{attn}^{b_j} \in \mathbb{R}^K$ for both items. Intuitively, the k-th element in $\mathbf{v}_{attn}^{t_i}/\mathbf{v}_{attn}^{b_j}$ indicates the performance of item t_i/b_j on a specific attribute a_k. By performing an outer product between those two vectors, we can enumerate over all the pairwise combinatorial effect between t_i and b_j's attribute-wise performance:

$$\mathbf{M}_{ij}^{affinity} = \mathbf{v}_{t_i}^{attn} \otimes \mathbf{v}_{b_j}^{attn} \in \mathbb{R}^{K \times K}, \tag{8}$$

where $\otimes$ is an outer product operator, and $\mathbf{M}_{ij}^{affinity} \in \mathbb{R}^{K \times K}$ is an explicit affinity matrix where a large element $m_{kk'}^{affinity} \in \mathbf{M}_{ij}^{affinity}$ indicates that t_i and b_j are respectively performing well on attributes a_k and a'_k, yielding a higher affinity score between their explicit attributes.

Then, a weighted compatibility score matrix $\mathbf{M}_{ij}^{weighted\_compat}$ for t_i and b_j can be obtained via an element-wise multiplication:

$$\mathbf{M}_{ij}^{weighted\_compat} = \mathbf{M}_{ij}^{affinity} \odot \mathbf{M}_{ij}^{compat} \in \mathbb{R}^{K \times K}, \tag{9}$$

where $\odot$ is an element-wise multiplication operator. Through element-wise multiplication, it is clear that a large score $m_{kk'} \in \mathbf{M}_{ij}^{weighted\_compat}$ can be obtained only if $m_{kk'}^{compact}$ and $m_{kk'}^{affinity}$ are both high. Hence, for t_i and b_j, a large $m_{kk'}$ means: (1) their attributes a_k and a'_k complement each other; and (2) t_i and b_j have ideal performance on a_k and a'_k, respectively. In this way, AFRec is able to give an explicit explanation on which pairs of attributes are most critical and have the most positive or negative impacts on the outfit. Furthermore, by incorporating fine-grained attribute-level affinity, we have higher chances of improving the recommendation accuracy, because the complementary information from different attribute views offers richer signals for identifying compatible clothes.

Eventually, the final compatibility score between t_i and b_j $\widehat{y}_{ij}^{compat}$ is a scalar derived by summing up all elements in $\mathbf{M}_{ij}^{weighted\_compat}$:

$$\widehat{y}_{ij}^{compat} = \sum_{k=1}^{K} \sum_{k'=1}^{K} m_{kk'}, \quad m_{kk'} \in \mathbf{M}_{ij}^{weighted\_compat}. \tag{10}$$

4.5 Learning objective

In a sense, only the positive top-bottom outfit pairs created by fashion experts are available in the dataset, while the negative pairs are unknown. Thus, we employ a soft-ranking loss function, namely Bayesian Personalised Ranking (BPR) [18] to explore the implicit relations between tops and bottoms. Specifically, for each observed top-bottom pair (t_i, b_j), we can generate two corrupted pairs $(t_i, b_{j'})$ and $(t_{i'}, b_j)$ by replacing either the top or bottom with an unobserved one. Then, based on the assumption that the observed pairs should be ranked higher than unobserved ones, we have:

$$\mathcal{L}_{bpr} = -\sum_{(i,j,j') \in \mathcal{D}} \ln \left(\sigma (\widehat{y}_{ij}^{compat} - \widehat{y}_{ij'}^{compat}) \right), \tag{11}$$

where $\mathcal{D}$ denotes all the training instances and σ is a sigmoid function. Note that we have omitted the corrupted instance for tops (i.e., (i', i, j)) to be succinct.

Ultimately, the final objective function of AFRec is formulated as follows:

$$\mathcal{L} = \mathcal{L}_{bpr} + \mathcal{L}_{category} + \mathcal{L}_{attribute}. \tag{12}$$

5 Experiments

To verify the effectiveness of our proposed model, we conduct extensive experiments on two real-world benchmark datasets, i.e., FashionVC and PolyvoreMaryland. We first describe the details of experimental settings and then give comprehensive analysis according to the experimental results.

5.1 Experimental settings

Datasets. We conduct experiments on two public fashion benchmark datasets, namely FashionVC and PolyvoreMaryland. **FashionVC**[3] is released by Song et al. [20], which consists of 20,726 outfits including 14,871 top item images and 13,663 bottom item images, created by fashion domain experts. Each clothing item in the dataset corresponds to an image, a text title and a category label. **PolyvoreMaryland**[4] is created by Han et al. [5], which has which has 21,889 outfits in total. We use images for visual information extraction, and characterise the attributes based on each item's title and category label. All the attributes and examples of their corresponding values are summarised in Table 1. We randomly split the data by 80%:10%:10% for training, validation and test, respectively.

5.2 Baseline methods

We compare our model AFRec with several state-of-the-art models for complementary clothing recommendation.

- **SiameseNet** [25]: The approach models compatibility by minimising the Euclidean distance between clothes pairs and making the incompatible ones far apart within a unified compatibility latent space through a contrastive objective function.
- **Monomer** [7]: The approach models fashion compatibility with a mixture of distances computed from multiple latent spaces.
- **BPR-DAE** [20]: The approach models the overall matching knowledge through an inner product of the top and bottom visual representations.
- **TripletNet** [1]: This is the state-of-the-art approach that captures the complementary relations among different fashion items with a triplet objective function.
- **TA-CSN** [23]: This is a category-aware method that considers item categorical awareness by generating category-specific masks added upon item visual embeddings, which helps the model to concentrate on different latent aspects when modelling items from different categories.
- **PAICM** [4]: It is the state-of-the-art attribute-aware approach that employs non-negative matrix factorisation to explore the pairwise compatibility at the attribute level.

[3] https://drive.google.com/file/d/1lO7M-jSWb25yucaW2Jj-9j_c9NqquSVF/view
[4] https://drive.google.com/drive/folders/0B4Eo9mft9jwoVDNEWlhEbUNUSE0

Table 1 A summarisation of all attributes and their corresponding values. We also report the attribute classification accuracy of AFRec on these attributes

Attribute	Attribute Values	Classification Accuracy (%)	
		FashionVC	Polyvore
Category	T-shirt, sweatshirts, cardigans sweaters, ...	98.2	87.3
Texture	cotton, fur, leather, velvet, metallic, ...	99.4	80.3
Style	patchwork, woven, slit, cuffed, sheer, raw, ...	99.4	84.6
Pattern	chino, houndstooth, striped, grid, crochet, ...	99.4	84.0
Neckline	scoop neck, v-neck, high-neck, tie-neck, ...	99.5	96.2
Sleeve Type	sleeveless, long sleeve, short sleeve, ...	98.8	92.3
Shape	oversized, stretch, skinny, loose, ...	97.7	73.1
Waistline	high waist, mid waist, low waist, ...	99.8	92.0
Bottom Leg	harem, straight-leg, cropped	99.9	95.2
Bottom Length	maxi, mini, midi	99.6	96.1

5.3 Evaluation protocols

For each positive top-bottom pair (t_i, b_j) in the test set, we generate negative test instances by replacing the bottom item b_j with 100 uniformly sampled negative bottom items that are not matched by the top item t_i. Then, we choose two commonly-used evaluation metrics, namely HR@K and Area Under the ROC Curve (AUC) to compare our model's performance against other baseline methods. HR@K indicates the percentage of correctly identified clothes pairs ranked in all top-K lists, which is formulated as the following:

$$\text{HR@}K = \frac{\#\text{hit@}K}{|D_{test}|}, \tag{13}$$

where D_{test} denotes the collection of all test cases. Meanwhile, AUC is defined as follows:

$$\text{AUC} = \frac{\sum pred_{positive} > pred_{negative}}{N_{positive} \times N_{negative}}, \tag{14}$$

where $\sum pred_{positive} > pred_{negative}$ represents the number of test cases that predicted score of positive pairs are larger than negative pairs, while $N_{positive}$ and $N_{negative}$ respectively denote the total number of positive and negative pairs.

5.4 Implementation details

AFRec is implemented using PyTorch [17] with Nvidia GTX 2080 Ti. For consistency, we apply the same dimension size D for all embeddings and hidden states. Specifically, we set D to 512. All the trainable parameters in our model are optimised using Adam optimiser [10] with the batch size of 64, the learning rate of 1e-4 and the weight decay of 1e-5. To help AFRec converge faster, we first pretrain the SAE module in AfRec using our attribute and category prediction objectives. The attribute classification accuracy of the pretrained SAE module is illustrated in Table 1. As can be seen, the model is highly confident in comprehensively extracting attribute information from visual features, and this allows AFRec to generate meaningful attribute-specific representations for the final recommendation task. After SAE is fully pretrained, we train the whole model in an end-to-end manner.

5.5 Analysis on recommendation effectiveness

We summarise the evaluation results of all models on the complementary clothing recommendation task with Table 2. From the results in the table, we can observe that our AFRec outperforms other state-of-the-art methods on most evaluation metrics, reflecting the effectiveness of our model. This is mainly because our model significantly benefits from the semantic attributes when modelling the compatibility at a fine-grained level. This helps AFRec better capture the complicated interactions among attributes. As a category-unaware model, SiameseNet merely learns fashion compatibility within a unified latent space, and it underperforms due to the lack of the ability to leverage the subtle yet informative attribute signals. By incorporating category-awareness in different learning schemes, we can observe similar performance from Monomer, BPR-DAE, Triplet Net and TA-CSN. This indicate that categorical information is helpful for advancing the performance in the task of complementary clothing recommendation. Among these methods, TA-CSN that uses type-specific mask embeddings yields better recommendation accuracy. This implies that instead of simply concatenating category embeddings to the global visual embeddings, performing mask operations can let the model focus on certain dimensions of item embeddings for downstream tasks. The attribute-aware method PAICM achieves similar performance to the category-aware methods, which demonstrates the effectiveness of incorporating attribute information for compatibility modelling. However, PAICM models compatibility with a single merged attribute-level embedding for each item. This modelling scheme may fail to capture sufficient disentangled attribute information since all attribute-specific information is fused. In contrast, our model not only accounts for the categorical information via categorical projection spaces, but also mines fine-grained compatibility relations by modelling meaningful semantic attribute interactions.

5.6 Ablation study

To verify the contribution of each proposed component in our model, we implement multiple variants of AFRec to perform an ablation study. The evaluation results on both datasets are

Table 2 Performance comparison between our proposed AFRec and other baseline methods

Methods	FashionVC					PolyvoreMaryland				
	AUC	HR@K				AUC	HR@K			
		K=5	K=10	K=20	K=40		K=5	K=10	K=20	K=40
SiameseNet	0.604	0.097	0.181	0.312	0.528	0.591	0.083	0.155	0.290	0.518
Monomer	0.702	**0.169**	0.286	0.458	0.691	0.705	0.176	0.289	0.457	0.690
BPR-DAE	0.709	0.167	0.273	0.467	0.704	0.695	0.173	0.282	0.439	0.675
Triplet Net	0.706	0.163	0.280	0.457	0.696	0.701	**0.181**	0.287	0.449	0.683
TA-CSN	0.716	0.167	0.284	0.467	0.708	0.702	0.173	0.284	0.451	0.684
PAICM	0.703	0.168	0.271	0.463	0.697	0.703	0.170	0.266	0.456	0.687
AFRec	**0.741**	0.164	**0.305**	**0.500**	**0.789**	**0.753**	0.180	**0.397**	**0.516**	**0.828**

demonstrated in Table 3. We introduce and analyse the effect of each variant of AFRec as follows:

- **AFRec**$_{w/o\_attr\_loss}$. This variant removes the attribute prediction loss, and all the embeddings of extracted attributes are treated as latent vectors containing different latent global visual information. The obvious performance drop on both datasets indicates that the attribute prediction task can effectively augment the expressiveness of the learned representations.
- **AFRec**$_{w/o\_cate\_loss}$. When removing the category classification loss, we can observe mild performance drop on both datasets. Intuitively, we use category classification loss to help the SAE module to concentrate on different parts of fashion items in different categories when learning their global visual features, making our reciprocal attention module highly effective.
- **AFRec**$_{w/o\_attention}$ and **AFRec**$_{self\_attention}$. We study the contribution of reciprocal attention module by either directly removing the whole attention module (i.e., AFRec$_{w/o_attention}$) or replacing it with a self-attention module (i.e., **AFRec**$_{self\_attention}$) that does not support attribute-wise comparison between different items. We can see similar performance drop on most evaluation metrics for these two variants. Hence, the results justify that our reciprocal attention effectively avoids the biases caused by the low compatibility scores of unimportant attribute features.
- **AFRec**$_{w/o\_cate\_projection}$. AFRec receives the worst evaluation results among all variants when its category-specific projection matrices are removed. This is mainly because the item compatibility measurement varies in different categories. The category-specific projection can let AFRec focus on different latent features of the attributes in different categories.
- **AFRec**$_{attr\_avg}$. This variant calculates the compatibility score using only a single embedding vector composed by averaging all attribute embeddings for each item. We can find a slight performance drop on both datasets. This is because the averaged attribute embeddings contain a mixture of multiple attribute characteristics, which may hinder AFRec from making precise compatibility measurement since all attribute factors are entangled.

Table 3 Performance comparison between different variants of AFRec

Variants	FashionVC				PolyvoreMaryland			
	AUC	HR@K			AUC	HR@K		
		K=10	K=20	K=40		K=10	K=20	K=40
AFRec$_{w/o_attr_loss}$ (3)	0.703	0.234	0.468	0.664	0.732	0.312	0.484	0.734
AFRec$_{w/o_cate_loss}$ (12)	0.717	0.281	0.461	0.688	0.750	0.344	0.508	0.773
AFRec$_{w/o_attention}$	0.703	0.172	0.461	0.703	0.749	0.344	0.515	0.758
AFRec$_{self_attention}$	0.718	0.227	0.445	0.703	0.740	0.367	0.492	0.758
AFRec$_{w/o_cate_projection}$ (7)	0.714	0.141	0.461	0.672	0.636	0.203	0.344	0.602
AFRec$_{attr_avg}$	0.731	0.258	0.453	0.688	0.724	0.305	0.477	0.727
Full Version	0.741	0.305	0.500	0.789	0.753	0.397	0.516	0.828

5.7 Analysis on recommendation explainability

5.7.1 Visualisation results

As attribute-wise compatibility learning plays a crucial role in facilitating the explainability of our model, we select four positive and negative pairs from FashionVC and Polyvore-Maryland dataset, respectively. We use four groups of examples, where each top item is paired with a successfully recommended bottom item and a negative item. We also visualise the computed weighted compatibility matrix $\mathbf{M}^{weighted\_compat}$ in Figure 4. Note that each entry in $\mathbf{M}^{weighted\_compat}$ is rescaled to [0, 1] range for better readability. For instance, in Figure 4(a), for the positive clothes pair, the values within the compatibility matrix are commonly higher than the negative clothes pair, indicating an overall strong complementary relationship between the grey sweater and the light blue jeans. To name a few, the key matching patterns for two items include the high compatibility between the textures of both items; also, the shape and sleeve type of the sweater are a good fit for the waistline design of the jeans recommended.

The second observation we can draw from this explainability study is that, for the same top item, its positive match (i.e., a bottom item) commonly performs better in almost all pairwise compatibility between attributes. Also, a positive item tends to exhibit advantages on some key attribute types over the negative item. For example, the compatibility between sleeve type (top) and waistline (bottom) Figure 4(b) varies significantly in positive and negative pairs. Similar results can be observed from the compatibility between pattern (top) and style (bottom) in Figure 4(c), and the compatibility between style (top) and waistline (bottom) in Figure 4(d). To summarise, the attribute-level explanation offers a highly intuitive means for users to understand the reasons why a pair of clothes are complementary or not. The explainability makes it easier to provide people with insights into which attribute factors are the main contributors in clothing matching.

5.7.2 User study

We further conduct a user study based on 10 randomly selected volunteers (5 are male and 5 are female) to quantitatively evaluate the utility of our generated explanations to real users. Specifically, we first use our model and the explainable baseline method PAICM [4] to generate the prediction and interpretation results for uniformly sampled 100 clothing outfits consisting of 50 positive and 50 negative pairs. In the user study, each participant are provided with all 100 visualisation results, and are asked to up-vote or down-vote the explanations generated by AFRec and PAICM. We collected responses from all participants, and report the up-vote ratio with Table 4. On the positive test instances, both methods can generate decent explanations on which attributes are critical when matching outfits, while AFRec still demonstrates more advantageous in the up-vote ratio. On negative instances, the explanations generated by AFRec are much more preferred. From the participants' responses, they mostly agree on the incompatible category, style, and texture attributes identified by AFRec. The key reason for better explainability of our model is that modelling interactions in an attribute-wise manner encourages the model to capture more details between two items. In comparison, PAICM merges the attribute information into a single embedding, which neglects the subtle information contained within the images, leading to unsatisfactory explanations.

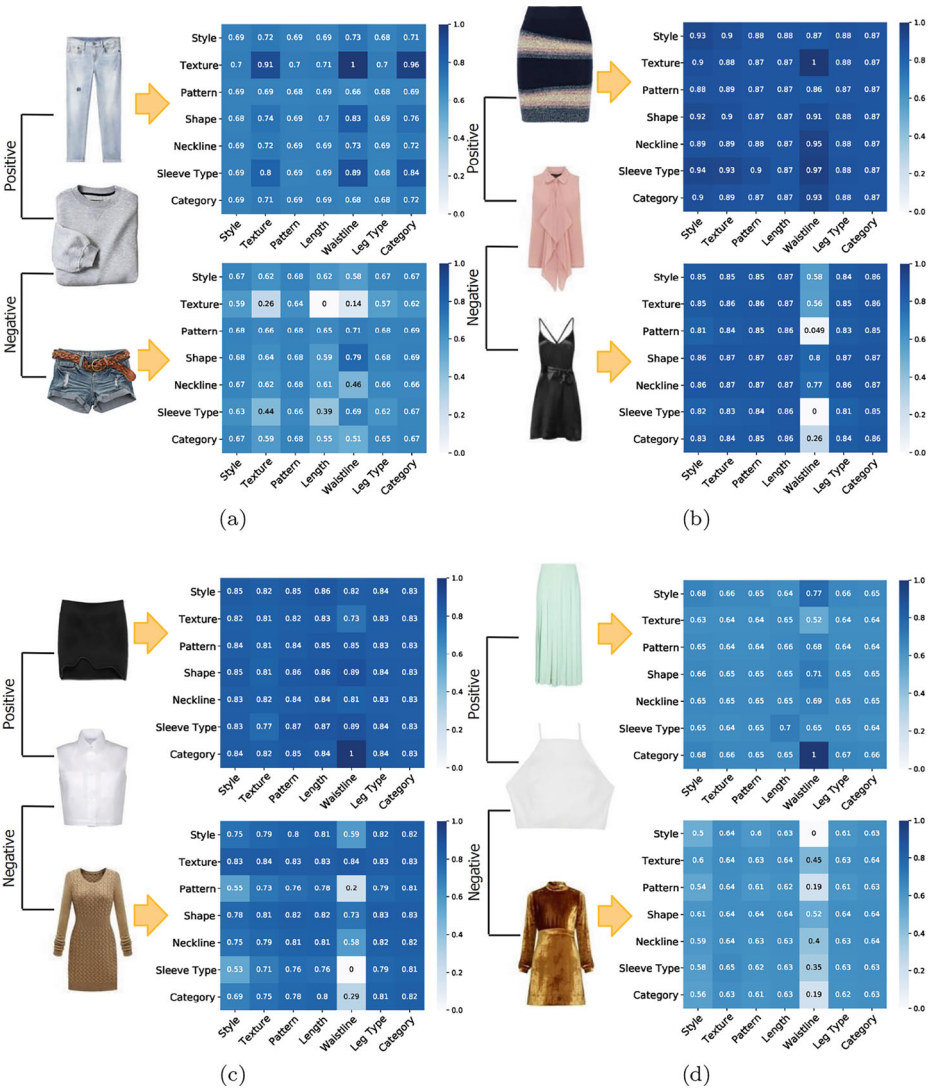

Figure 4 Visualisation of four pairs of positive and negative outfit test instances

Table 4 Up-vote rate of the generated explanations

	Model	Positive	Negative
Up-vote Ratio (%)	AFRec	**66.0**	**48.0**
	PAICM	64.0	38.0

6 Conclusion

To deal with the lack of explainability of existing complementary clothing recommendation approaches, we propose a novel solution named AFRec in this paper. AFRec obtains attribute-specific representations from fashion items by a CNN-based attribute embedding extractor to support fine-grained fashion compatibility modelling and enhances its explainability towards the prediction results. Our experiments on two large-scale benchmark datasets show the effectiveness and interpretability of AFRec, demonstrating the strong practicality in real-life scenarios.

References

1. Chen, L., He, Y.: Dress fashionably: Learn fashion collocation with deep mixed-category metric learning. In: AAAI, pp. 2103–2110 (2018)
2. Chiu, T.: Understanding generalized whitening and coloring transform for universal style transfer. In: 2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019, pp. 4451–4459. IEEE (2019)
3. Han, K., Guo, J., Zhang, C., Zhu, M.: Attribute-aware attention model for fine-grained representation learning. In: Boll, S., Lee, K.M., Luo, J., Zhu, W., Byun, H., Chen, C.W., Lienhart, R., Mei, T. (eds.) ACM MM 2040-2048, ACM (2018)
4. Han, X., Song, X., Yin, J., Wang, Y., Nie, L.: Prototype-guided attribute-wise interpretable scheme for clothing matching. In: Piwowarski, B., Chevalier, M., Gaussier, É., Maarek, Y., Nie, J., Scholer, F. (eds.) SIGIR, pp. 785–794. ACM (2019)
5. Han, X., Wu, Z., Jiang, Y., Davis, L.S.: Learning fashion compatibility with bidirectional lstms. In: MM, pp. 1078–1086 (2017)
6. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016, pp. 770–778. IEEE Computer Society (2016)
7. He, R., Packer, C., McAuley, J.J.: Learning compatibility across categories for heterogeneous item recommendation. In: ICDM, pp. 937–942 (2016)
8. Jetley, S., Lord, N.A., Lee, N., Torr, P.H.S.: Learn to pay attention. In: ICLR. OpenReview.net (2018)
9. Kang, W.C., Kim, E., Leskovec, J., Rosenberg, C., McAuley, J.: Complete the look: Scene-based complementary product recommendation. In: CVPR, pp. 10532–10541 (2019)
10. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: ICLR (2015)
11. Li, Y., Cao, L., Zhu, J., Luo, J.: Mining fashion outfit composition using an end-to-end deep learning approach on set data. Trans. MM **19**(8), 1946–1955 (2017)
12. Li, Y., Luo, Y., Huang, Z.: Graph-based relation-aware representation learning for clothing matching. In: Borovica-Gajic, R., Qi, J., Wang, W. (eds.) ADC, Lecture Notes in Computer Science, vol. 12008, pp. 189–197. Springer (2020)
13. Lin, Y., Ren, P., Chen, Z., Ren, Z., Ma, J., de Rijke, M.: Explainable outfit recommendation with joint outfit matching and comment generation. TKDE **32**(8), 1502–1516 (2020)
14. Liu, S., Feng, J., Domokos, C., Xu, H., Huang, J., Hu, Z., Yan, S.: Fashion parsing with weak color-category labels. TMM **16**(1), 253–265 (2014)
15. Liu, Z., Luo, P., Qiu, S., Wang, X., Tang, X.: Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In: CVPR, pp. 1096–1104 (2016)
16. McAuley, J.J., Targett, C., Shi, Q., van den Hengel, A.: Image-based recommendations on styles and substitutes. In: SIGIR, pp. 43–52 (2015)
17. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Köpf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S.: Pytorch: An imperative style, high-performance deep learning library. In: Wallach, H.M., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E.B., Garnett, R. (eds.) NIPS, pp. 8024–8035 (2019)
18. Rendle, S., Freudenthaler, C., Gantner, Z., Schmidt-Thieme, L.: BPR: bayesian personalized ranking from implicit feedback. In: Bilmes, J.A., Ng, A.Y. (eds.) UAI, pp. 452–461. AUAI Press (2009)

19. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M.S., Berg, A.C., Li, F.: Imagenet large scale visual recognition challenge. Int. J. Comput. Vis. **115**(3), 211–252 (2015)
20. Song, X., Feng, F., Liu, J., Li, Z., Nie, L., Ma, J.: Neurostylist: Neural compatibility modeling for clothing matching. In: ACM MM, pp. 753–761 (2017)
21. Song, X., Nie, L., Wang, Y.: Compatibility modeling: Data and knowledge applications for clothing matching. Synth. Lect. Inf. Concepts Retr. Serv. **11**(3), 1–138 (2019)
22. Tangseng, P., Yamaguchi, K., Okatani, T.: Recommending outfits from personal closet. In: WACV, pp. 269–277 (2018)
23. Vasileva, M.I., Plummer, B.A., Dusad, K., Rajpal, S., Kumar, R., Forsyth, D.A.: Learning type-aware embeddings for fashion compatibility. In: ECCV, pp. 405–421 (2018)
24. Veit, A., Belongie, S.J., Karaletsos, T.: Conditional similarity networks. In: CVPR, pp. 1781–1789 (2017)
25. Veit, A., Kovacs, B., Bell, S., McAuley, J.J., Bala, K., Belongie, S.J.: Learning visual clothing style with heterogeneous dyadic co-occurrences. In: ICCV, pp. 4642–4650 (2015)
26. Yang, X., He, X., Wang, X., Ma, Y., Feng, F., Wang, M., Chua, T.: Interpretable fashion matching with rich attributes. In: SIGIR, pp. 775–784 (2019)
27. Yang, X., Ma, Y., Liao, L., Wang, M., Chua, T.: Transnfcm: Translation-based neural fashion compatibility modeling. In: AAAI, pp. 403–410 (2019)
28. Zhang, Y., Zhang, P., Yuan, C., Wang, Z.: Texture and shape biased two-stream networks for clothing classification and attribute recognition. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020, pp. 13535–13544. IEEE (2020)
29. Zhou, B., Khosla, A., Lapedriza, À., Oliva, A., Torralba, A.: Learning deep features for discriminative localization. In: CVPR, pp. 2921–2929. IEEE Computer Society (2016)