Prototype-guided Attribute-wise Interpretable Scheme for Clothing Matching

Xianjing Han
Shandong University
hanxianjing2018@gmail.com

Xuemeng Song∗
Shandong University
sxmustc@gmail.com

Jianhua Yin
Shandong University
jhyin@sdu.edu.cn

Yinglong Wang
Shandong Computer Science Center
(National Supercomputer Center in Jinan)
wangyl@sdas.org

Liqiang Nie∗
Shandong University
nieliqiang@gmail.com

ABSTRACT
Recently, as an essential part of people’s daily life, clothing matching has gained increasing research attention. Most existing efforts focus on the numerical compatibility modeling between fashion items with advanced neural networks, and hence suffer from the poor interpretation, which makes them less applicable in real world applications. In fact, people prefer to know not only whether the given fashion items are compatible, but also the reasonable interpretations as well as suggestions regarding how to make the incompatible outfit harmonious. Considering that the research line of the comprehensively interpretable clothing matching is largely untapped, in this work, we propose a prototype-guided attribute-wise interpretable compatibility modeling (PAICM) scheme, which seamlessly integrates the latent compatible/incompatible prototype learning and compatibility modeling with the Bayesian personalized ranking (BPR) framework. In particular, the latent attribute interaction prototypes, learned by the non-negative matrix factorization (NMF), are treated as templates to interpret the discordant attribute and suggest the alternative item for each fashion item pair. Extensive experiments on the real-world dataset have demonstrated the effectiveness of our scheme.

CCS CONCEPTS
• Information systems → Retrieval tasks and goals; World Wide Web;

KEYWORDS
Fashion Analysis, Interpretable Compatibility Modeling, Non-negative Matrix Factorization.

* Xuemeng Song (sxmustc@gmail.com) and Liqiang Nie (nieliqiang@gmail.com) are corresponding authors.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

ACM Reference Format:

1 INTRODUCTION

Nowadays, clothing matching has become an indispensable part of people’s daily life, since a properly coordinated outfit can improve one’s appearance greatly. However, not everyone is a natural-born fashion stylist, and for those who lack the taste of aesthetics, matching clothes and making proper outfits has become their daily headache. Therefore, it is thus highly desirable to devise an automatic clothing matching scheme to aid people in outfit composition. Towards this end, three essential questions frequently faced by people in clothing matching merit our special attention. As shown in Figure 1, Q1: Whether the given fashion items are compatible? Q2: What are the discordant components that result in the incompatible matching? Q3: What are the alternative items to transform the incompatible pairs to compatible ones? In fact, the recent proliferation of many online fashion communities, such as IQON1 and Chictopia2, contributing a large number of outfits composed by fashion experts, has enabled researchers to tackle the automatic clothing matching problem. Due to their huge success in various domains, most of existing efforts employ deep learning methods to learn effective representations of fashion items, based on that they can measure the compatibility between fashion items.

1http://www.iqon.jp/.
2http://www.chictopia.com/.
Nevertheless, as pure data-driven learning schemes, deep learning methods suffer from the poor interpretability given that each dimension of the learned representation cannot explicitly refer to an intuitive semantic aspect of fashion items, causing the questions Q2 and Q3 of requiring more result interpretations largely untapped. Notably, although a few pioneer researchers have attempted to tackle the question Q2 by enhancing the interpretability through modeling the attribute-level (e.g., color and texture) compatibility between fashion items [7], they cannot provide the comprehensive interpretation due to the extremely limited attributes they adopt.

In this work, we aim to comprehensively tackle all the three essential problems, namely, the compatibility determination between fashion items, discordant component interpretation for incompatible outfits, and alternative item suggestion towards making compatible ones. We focus on devising a versatile attribute-wise interpretable clothing matching scheme, since attributes are the most intuitive semantic cues to characterize fashion items. However, fulfilling the task in the attribute-wise manner is non-trivial due to the following challenges. 1) As aforementioned, attribute plays a pivotal role in both characterizing fashion items and interpreting the matching results. However, most of existing benchmark datasets pertaining to clothing matching lack the attribute ground truth for fashion items. How to acquire the accurate fine-grained attribute representations for the benchmark datasets poses a primary challenge for us. 2) As the saying goes, things of one kind come together. Compatible fashion items may essentially follow certain underlying harmonious attribute interaction prototypes, while the incompatible ones would also share several unfavorable attribute compositions. For example, \{chiffon, pear-shaped, garden, beadings\} tends to be a harmonious attribute interaction prototype, while \{boyfriend-style, silk lace gauze, active wear, floral printing\} can be an incompatible one. Therefore, how to explore the latent compatible/incompatible attribute interaction prototypes and fashion items can be adaptively coupled constitutes another challenge.

To address the aforementioned challenges, we propose a prototype-guided attribute-wise interpretable compatibility modeling scheme, termed PAICM, to jointly regularize the latent prototype learning and compatibility modeling, as shown in Figure 2. Without losing the generality, here we study the problem of clothing matching between tops and bottoms. In particular, to facilitate the matching result interpretation, the scheme first extracts the semantic attribute representations for fashion items with a set of advanced neural networks, where each network is aligned to an attribute to ensure the quality of the attribute representation. Notably, to enhance the portability of PAICM, apart from our primary dataset adopted for clothing matching, we introduce an auxiliary dataset of fashion items with rich attribute annotations to pre-train the attribute classification networks. Based on the learned attribute representations, on one hand, the proposed scheme explores the latent compatible and incompatible attribute interaction prototypes using the non-negative matrix factorization (NMF) [17]. The learned prototypes are regarded as the templates to guide the discordant attribute interpretation and the alternative item suggestion. On the other hand, towards compatibility modeling, the proposed scheme seeks the latent space to accurately measure the compatibility between fashion items using the multi-layer perceptron (MLP). Ultimately, the proposed scheme seamlessly integrates the latent prototype learning and compatibility modeling with the Bayesian personalized ranking (BPR) framework [31], where the pairwise preferences between attribute prototypes and fashion items can be adaptively coupled and well exploited.

Our main contributions can be summarized in threefold:

- To the best of our knowledge, this is the first attempt to comprehensively fulfill the automatic clothing matching task by answering the three essential questions of the compatibility determination, discordant component interpretation, and alternative item suggestion. We propose a prototype-guided attribute-wise interpretable compatibility modeling scheme PAICM, where the latent compatible and incompatible prototype learning and compatibility modeling is jointly regularized.

- Extensive experiments have been conducted on the real-world dataset, which demonstrates the effectiveness of the proposed scheme. As a byproduct, we released the codes, and involved parameters to benefit other researchers.

The remainder of this paper is structured as follows. Section 2 briefly reviews the related work. In Section 3, we detail the proposed model. We present the experimental results and analyses in Section 4, followed by our concluding remarks and future work in Section 5.

2 RELATED WORK

2.1 Fashion Analyses

In recent years, the huge economic value of the fashion market has attracted many researchers’ attention. Increasing efforts have been dedicated to the fashion domain, such as the fashionability prediction [19, 33], fashion trending prediction [8, 46], clothing retrieval [12, 24, 25] and compatibility modeling [9, 34, 42]. For example, McAuley et al. [28] proposed a general framework to model the human visual preference for a given pair of objects based on the Amazon real-world co-purchase dataset. In addition, Song et al. [35] investigated the problem of complementary fashion item matching with a multi-modal dataset collected from Polyvore4. Later, Lin et al. [20] further explored the user comments to boost the performance of fashion item recommendation, where a more comprehensive dataset ExpFashion was introduced. Although existing researches have achieved compelling success, they mainly focused on utilizing deep learning methods to represent fashion items with the blurry semantic features, resulting in their poor interpretability. To enhance the model interpretability, Feng et. al. [7] proposed a partition embedding network to learn

---

1https://anonymity2019.wixsite.com/paicm/

2Polyvore has been acquired by the global fashion platform Ssense in 2018.
the embedding of each attribute and then model the attribute-level compatibility between fashion items. Despite the promising performance it accomplished, the attributes regarding the compatibility of fashion items can be numerous yet they only adopted limited ones, making the interpretation incomplete. Distinguished from these studies, we aim to not only improve the interpretability of the clothing matching in a comprehensive attribute-wise manner but also facilitate the alternative item suggestion.

2.2 Matrix Factorization
As a numerical analysis method, matrix factorization (MF) is widely applied in various research areas, such as the item recommendation [11, 15, 38, 40] and information retrieval [26, 30, 37], due to its superior performance in discovering the latent features between two entities (e.g., the user and item). In order to effectively adapt to different tasks, several variants of MF have been devised, such as the singular value decomposition (SVD) [6], probabilistic matrix factorization (PMF) [29] and non-negative matrix factorization (NMF) [17], and their efficiency has been validated in various domains. For example, Sun et al. [36] proposed a SVDNet to fulfill the retrieval task of person re-identification (reID), where the SVD is employed for the optimization of the deep representation learning process. In addition, Kim et al. [14] presented a context-aware convolutional matrix factorization (ConvMF) that integrates the convolutional neural network (CNN) into the PMF in the context of document context-aware recommendation. Besides, as a useful tool for the sparse and meaningful feature extraction, NMF also drew researchers’ attention. For example, Xu et al. [41] proposed a document clustering method based on the NMF with the term-document matrix. Furthermore, to forecast the fashion styles, Ziad et. al. [46] employed the NMF to discover the latent clothing styles in an unsupervised manner. Although the NMF has been successfully applied to solve tasks like text clustering [41], fashion trending prediction [46] and recommender systems [1], limited efforts have been dedicated to the complementary clothing matching, which is the major concern of our work.

3 METHODOLOGY
In this section, we first formally give the research problem formulation, and then detail the proposed PAICM.

3.1 Problem Formulation
Formally, we first declare some notations used in this work. We use bold capital letters (e.g., \(X\)) and bold lowercase letters (e.g., \(x\)) to denote matrices and vectors, respectively. We employ non-bold letters (e.g., \(x\)) to represent scalars and Greek letters (e.g., \(\beta\)) to denote the parameters. If not clarified, all vectors are in the column forms. \(\|A\|_F\) denotes the Frobenius norm of matrix \(A\).

In the real-world clothing matching scenarios, users may not only want to know whether the given fashion items are compatible or not, but also expect to get advice on how to harmonize the improper outfit. In this context, we aim to devise an attribute-wise interpretable compatibility modeling scheme to explain the underlying reasons why the given items are incompatible in the attribute-wise manner and provide the potential attribute manipulations to make compatible outfits. Assume that we have a set of tops \(T = \{t_1, t_2, \ldots, t_{N_t}\}\) and bottoms \(B = \{b_1, b_2, \ldots, b_{N_b}\}\), where \(N_t\) and \(N_b\) denote the total number of tops and bottoms,
respectively. Each item \( t_i \) (\( b_j \)) is associated with an image with a clear background, the textual description and structured category labels. In this work, we characterize each fashion item with a set of attributes (e.g., color and category) \( \mathcal{A} = \{a_q\}_{q=1}^Q \), where \( a_q \) is the \( q \)-th attribute and \( Q \) is the total number of attributes. Each attribute \( a_q \) is associated with a set of elements representing its possible values \( \mathcal{E}_q = \{e_{q1}, e_{q2}, \ldots, e_{qM_q}\} \), where \( e_{qk} \) refers to the \( i \)-th element and \( M_q \) is the total number of elements regarding \( a_q \). For simplicity, we compile all \( \mathcal{E}_q \)'s in order and hence derive a unified set of attribute elements \( \mathcal{E} = \bigcup_{q=1}^Q \mathcal{E}_q = \{e_1, e_2, \ldots, e_M\} \), where \( M = \sum_{q=1}^Q M_q \). In addition, we have a set of positive top-bottom pairs \( S = \{(t_i, b_j), (t_i, b_j), \ldots, (t_n, b_j)\} \) composed by fashion experts, where \( N \) is the total number of positive pairs. Accordingly, for each top \( t_i \), we can derive a set of positive bottoms \( B^+_i = \{b_j \in B(t_i, b_j) \in S\} \). Let \( s_{ij} \) denote the compatibility between the top \( t_i \) and bottom \( b_j \) based on which we can distinguish whether the given fashion items are compatible or not.

### 3.2 Semantic Attribute Representation

As a matter of fact, the online fashion item is usually characterized by a visual image, certain user-generated textual description and structured category labels. In a sense, the visual image and structured category labels can faithfully capture the essential features of fashion items, such as the color, shape and category, whereas the user-generated textual description may be unreliable as it can be intrinsically noisy, not to mention the mendacious ones edited by crafty sellers. Therefore, similar to the existing work [46], we only exploit the reliable visual cues as well as the structured category information to model the compatibility between fashion items. Notably, existing efforts mainly adopt advanced deep neural networks to learn the effective presentations for fashion items and measure the compatibility owing to their compelling success in various research tasks. Nevertheless, as a pure data-driven learning scheme, deep neural network suffers from the poor interpretability due to the fact that each dimension of the learned representation cannot explicitly refer to the intuitive semantic aspect of fashion items. Towards this end, we aim to learn the meaningful representations for fashion items, whose dimensions directly stand for the semantic attributes and hence enhance the model interpretability.

On one hand, regarding the sophisticated visual signals, we argue that taking advantage of the well pre-trained attribute classification networks is the most natural and straightforward way to obtain the interpretable semantic representations of fashion items. As to ensure the performance of the attribute classification networks, we align each attribute \( a_q \) with a separate attribute classification network \( h_q \). It is worth noting that as the category information also contributes an essential attribute of fashion items, here we have \( Q - 1 \) attributes characterized by the visual cues. We feed the visual image \( I_t \) of the \( i \)-th top/bottom into these \( h_q \)'s, and obtain the semantic attribute representations as follows,

\[
\begin{align*}
t^q_i &= h_q(I_t; \Theta_q), \quad q = 1, 2, \ldots, Q - 1, \tag{1}
\end{align*}
\]

where \( \Theta_q \) denotes the network parameter of \( h_q \) and \( t^q \in \mathbb{R}^{M_q} \) is the network output of \( h_q \). The \( d \)-th entry in \( t^q_i \) refers to the probability that the top \( t_i \) presents the attribute element \( e_{qk} \). In particular, we denote \( \mathbf{t}_i^q = [t^q_1; t^q_2; \ldots; t^q_{Q-1}] \) as the final semantic attribute representation of the \( i \)-th top/bottom derived from the visual signals, where \( \cdot \) is the cascading operation of vectors in the vertical direction.

On the other hand, the intuitive nature of the structured category information propels us to encode it directly with the one-hot representation. Let \( \mathbf{f}_i^q \) stands for the one-hot semantic attribute representation derived from the category context for the \( i \)-th top/bottom. Ultimately, we concatenate the attribute representations obtained from both sources and generate the final semantic attribute representation \( \mathbf{f}_i = [\mathbf{f}_i^q; \mathbf{f}_i^q] \) for the \( i \)-th item.

#### 3.3 Latent Compatibility Space

Apparently, it is not advisable to directly measure the compatibility in the raw attribute space. Similar to [34], we assume that there is a latent compatibility space that enables us to accurately model the complicated attribute interactions and hence boost the compatibility modeling performance. In this work, we resort to the MLP, which has shown superior performance in various representation learning tasks [21–23, 39]. In particular, we add \( K \) hidden layers over the semantic attribute representation of the fashion item as follows,

\[
\begin{align*}
\mathbf{t}^q_i &= \mathbf{f}^q_i, \\
\mathbf{f}^q_i &= \mathbf{f}^{q-1}_i, \\
\mathbf{f}^{q-1}_i &= \sigma(W^q_i \mathbf{t}^q_i + b^q_i), \quad k = 1, \ldots, K, \quad y \in \{t, b\}, \tag{2}
\end{align*}
\]

where \( \mathbf{f}^q_i \) is the \( k \)-th layer hidden representation, \( W^q_i \) and \( b^q_i \) are weight matrices and biases, respectively. \( t \) and \( b \) denote top and bottom. \( \sigma : \mathbb{R} \mapsto \mathbb{R} \) is a non-linear function applied in an element-wise manner, where we choose the sigmoid function \( \sigma(x) = \frac{1}{1 + \exp(-x)} \) in this work. The latent representation of the fashion item is defined as the output of the \( K \)-th layer, i.e., \( \mathbf{f}_i^K = \mathbf{f}^K_i \in \mathbb{R}^{D_1}, y \in \{t, b\} \), where \( D_1 \) denotes the dimension of the latent compatibility space. Therefore, the compatibility between top \( t_i \) and bottom \( b_j \) can be measured as follows,

\[
s_{ij} = \langle \mathbf{t}_i^T \mathbf{f}_j \rangle. \tag{3}
\]

In a sense, we can assume that the top-bottom pairs composed by fashion experts are the positive (compatible) ones. However, it may be too absolute to claim that the other fashion item pairs are negative (incompatible), since they can be the potential positive ones whose items may be paired later. In order to model the implicit relations between tops and bottoms, we adopt the BPR framework for its excellent performance on the implicit preference modeling [4, 11]. In particular, we argue that as for top \( t_i \), bottoms in the positive set \( B^+_i \) are more compatible than the other bottoms. Accordingly, we construct the training set \( \mathcal{D}_t = \{(i, j, k) \in T, b_j \in B^+_i \land b_k \in B \setminus B^+_i\} \), where the triplet \((i, j, k)\) indicates that top \( t_i \) goes better with bottom \( b_j \) as compared with bottom \( b_k \). According to [31], the objective function can be written as follows,

\[
L^\text{item}_{\text{bpr}} = \sum_{(i, j, k) \in \mathcal{D}_t} -\log(\sigma(s_{ij} - s_{ik})) + \frac{\lambda}{2} \|\Omega\|_F^2, \tag{4}
\]

where \( \sigma \) is the sigmoid function, \( \lambda \) is the non-negative hyperparameter to avoid the overfitting and \( \Omega \) denotes the set of parameters (i.e., \( W^q_i \)'s and \( b^q_i \)'s).
### Algorithm 1 Prototype-guided Compatibility Modeling.

**Input:** \( \mathcal{D}_S = \{(i, j, k)\}, \) \( \mu, \nu, L_p, L_u \)

**Output:** Parameters \( \Omega \) in MLP, parameters \( P, H_p, U \) and \( H_u \) in NMF.

1. Initialize neural network parameters in MLP and NMF.
2. **repeat**
3. Randomly draw \((i, j, k)\) from \( \mathcal{D}_S \)
4. Calculate \( l^* \) and \( r^* \) according to Eqn. (10).
5. Update \( \Omega, P, H_p, U \) and \( H_u \) according to Eqn. (12).
6. **until** Converge
7. Identify the discordant attribute \( a^* \) for the given negative top-bottom pair according to Eqn. (16)
8. Manipulate the discordant attribute representation and retrieve the new fashion item.

In the same manner, we can also derive the latent incompatible prototypes based on the set of negative top-bottom pairs \((t_i, b_j)\)'s, where the bottom \( b_j \) is randomly sampled for top \( t_i \). Let \( G_u \in \mathbb{R}^{L \times N} \) be the data matrix comprising semantic attribute representations of negative top-bottom pairs and \( U = [u_1, u_2, \cdots, u_n] \in \mathbb{R}^{L \times N} \) be the matrix of latent incompatible prototypes, where \( L_u \) is the total number of incompatible prototypes, and \( H_u \in \mathbb{R}^{L_u \times N} \) denotes the latent representation matrix of the \( N \) negative top-bottom pairs in the prototype space. Similarly, we represent the \( r \)-th latent incompatible prototype \( u_r \in \mathbb{R}^M \) as follows,

\[
    u_r = \frac{u_f^r}{\|u_f^r\|_2}, \quad u_b^r \quad \text{subject to} \quad \|u_f^r\|_2 = \|u_b^r\|_2 = 1, \quad \frac{u_f^r}{\|u_f^r\|_2} \cdot \frac{u_b^r}{\|u_b^r\|_2} = 0,
\]

where \( u_f^r \in \mathbb{R}^M \) and \( u_b^r \in \mathbb{R}^M \) denote the semantic attribute representations of the prototype top and bottom of \( u_r \). Therefore, we have the following NMF loss for the latent prototype learning,

\[
    \mathcal{L}_{nmf} = \|G_p - PH_p\|_F^2 + \|G_u - UH_u\|_F^2. \quad (8)
\]

It is intuitive that the top and bottom of one compatible prototype should be more compatible than those of the incompatible ones. Therefore, we define the intrinsic compatibility for each prototype \( p_l(u_r) \) as follows,

\[
    s_f^r = \langle \tilde{u}_f^r, P \rangle, \quad s_b^r = \langle \tilde{u}_b^r, P \rangle \quad (9)
\]

where \( s_f^r \) and \( s_b^r \) are the intrinsic compatibility for the compatible prototype \( p_l \) and incompatible prototype \( u_r \), respectively. \( \tilde{u}_f^r \) and \( \tilde{u}_b^r \) are the hidden representations of \( p_l \) and \( u_r \), respectively, which can be acquired based on Eqn. (2).

To seamlessly integrate the latent prototype learning and compatible modeling, for each sample \((i, j, k)\), we particularly define its most similar compatible and incompatible prototypes \( p_l^* \) and \( u_r^* \) with the Euclidean distance, whose indexes \( l^* \) and \( r^* \) can be derived as follows,

\[
    \begin{align*}
        &d_{p}((i, j, l), (i, j, l)) = \left\| \frac{p_f^l}{\|p_f^l\|_2} - \frac{p_f^{' l}}{\|p_f^{' l}\|_2} \right\|_2, & d_{u}(i, k, r) = \left\| \frac{u_f^r}{\|u_f^r\|_2} - \frac{u_f^{' r}}{\|u_f^{' r}\|_2} \right\|_2, \\
        &l^* = \arg \min_{l} d_{p}(i, j, l), & r^* = \arg \min_{r} d_{u}(i, k, r).
    \end{align*}
\]

In a sense, we expect that the intrinsic compatibility of the compatible prototype \( p_l^* \) should be higher than that of the...
incompatible one \( u_r \). Therefore according to the BPR, we thus have the following adaptive objective function,

\[
\mathcal{L}^{proto}_{bpr} = \sum_{(i,j,k) \in D_s} -\ln(\sigma(t^P_{ik} - s^u_{ir})),
\]

where \( s^P_i \) and \( s^u_i \) can be obtained with Eqn. (9). Interestingly, with \( x^{item}_{bpr} \) and \( x^{proto}_{bpr} \), the compatibility modeling between fashion items and the prototype learning can be mutually promoted. Ultimately, we obtain the final objective function as follows,

\[
\mathcal{L} = x^{item}_{bpr} + \mu x^{proto}_{bpr} + \nu \mathcal{L}_{nmf},
\]

where \( \mu \) and \( \nu \) are the non-negative trade-off hyperparameters to weigh the different components of the objective function.

### 3.5 Interpretable Attribute Manipulation

In order to transform the incompatible fashion item pairs into the compatible ones, we first employ the \( L_2 \) compatible prototypes as templates to identify the discordant attributes. In particular, for the given negative (incompatible) top-bottom pair \((t_i, b_k)\), we particularly find the most similar compatible prototype \( p_t \), according to Eqn. (10). For simplicity, we divide \( p_t \) into \( Z \) parts as follows,

\[
p_t = \left[ p^1_t; \cdots ; p^Q_t; O^{Q+1}; \cdots ; p^Z_t \right].
\]

where \( Z = 2Q \). The first \( Q \) parts refer to the attribute representations of the top in prototype \( p_t \), while the last \( Q \) parts correspond to that of the bottom in \( p_t \). In the same manner, the negative top-bottom pair \((t_i, b_k)\) can be represented as follows,

\[
g_{ik} = \left[ f^1_{ik}; f^Q_{ik}; \cdots ; f^Z_{ik} \right]. \quad (14)
\]

Moreover, we define the attribute-wise difference between \((t_i, b_k)\) and \( p_t \) as follows,

\[
d_c(i, k, l^*, z) = \frac{\| g_{ikl} - P^*_{z} \|}{M_z},
\]

where \( d_c(i, k, l^*, z) \) denotes the attribute difference between \((t_i, b_k)\) and \( p_t \) regarding the \( z \)-th attribute. We then identify the most discordant attribute that causes the incompatibility as follows,

\[
z^* = \arg \max_z d_c(i, k, l^*, z). \quad (16)
\]

Thereafter, to suggest the alternative item and make the compatible pair, we replace the attribute representation \( g_{ik}^* \) of

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Top</th>
<th>Trousers</th>
<th>Dress</th>
</tr>
</thead>
<tbody>
<tr>
<td>length of upper-body clothes</td>
<td>0.7606</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>type of trousers</td>
<td>-</td>
<td>0.7233</td>
<td>-</td>
</tr>
<tr>
<td>part details of clothes</td>
<td>0.8462</td>
<td>0.8697</td>
<td>0.8181</td>
</tr>
<tr>
<td>type of clothes buttons</td>
<td>0.6742</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>length of trousers</td>
<td>-</td>
<td>0.7707</td>
<td>-</td>
</tr>
<tr>
<td>style of clothes</td>
<td>0.7698</td>
<td>0.7575</td>
<td>0.8325</td>
</tr>
<tr>
<td>fabric of clothes</td>
<td>0.8117</td>
<td>0.8738</td>
<td>0.8241</td>
</tr>
<tr>
<td>type of waistlines</td>
<td>-</td>
<td>0.8171</td>
<td>0.7798</td>
</tr>
<tr>
<td>texture of clothes</td>
<td>0.7668</td>
<td>0.8170</td>
<td>0.7387</td>
</tr>
<tr>
<td>graphic elements of clothes</td>
<td>0.7433</td>
<td>0.8166</td>
<td>0.7741</td>
</tr>
<tr>
<td>length of dresses</td>
<td>-</td>
<td>-</td>
<td>0.8243</td>
</tr>
<tr>
<td>design of dresses</td>
<td>-</td>
<td>-</td>
<td>0.8446</td>
</tr>
<tr>
<td>length of sleeves</td>
<td>0.7975</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>fitness of clothes</td>
<td>0.7135</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>type of collars</td>
<td>0.7839</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>type of trousers</td>
<td>-</td>
<td>-</td>
<td>0.7694</td>
</tr>
<tr>
<td>thickness of clothes</td>
<td>0.7668</td>
<td>0.8126</td>
<td>-</td>
</tr>
<tr>
<td>type of sleeves</td>
<td>0.7219</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

\((t_i, b_k)\) with \( P^*_{z} \) and hence obtain the manipulated semantic attribute representation as follows,

\[
\tilde{g}_{ik} = \begin{cases} f^1_{ik} ; f^Q_{ik} ; \cdots ; f^Z_{ik} & \text{if } z^* \leq Q, \\
\tilde{f}^1_{ik} ; \tilde{f}^Q_{ik} ; \cdots ; \tilde{f}^Z_{ik} & \text{if } z^* > Q, \end{cases}
\]

where \( f^1_{ik} \) and \( \tilde{f}^Q_{ik} \) are the manipulated semantic attribute representation of top \( t_i \) and bottom \( b_k \), respectively, using which we can retrieve the new fashion items to make a compatible matching. In particular, if the discordant attribute manipulation needs to be taken on the top \( t_i \) (i.e., \( z^* \leq Q \)), we can retrieve new tops \( t_j \)'s by ranking the Euclidean distance \( d_{p_t} \)'s between \( f^1_{ik} \) and the semantic attribute representations of training tops in the decent order. Otherwise, we can retrieve new bottoms \( b_k \)'s by ranking \( d_{p_t} \)'s between \( f^Q_{ik} \) and the representations of training bottoms. The workflow of attribute manipulation is shown in Figure 3, and the algorithm of the proposed method is summarized in Algorithm 1.

### 4 EXPERIMENT

To validate the effectiveness of the proposed model, we conducted extensive experiments on the real-world dataset FashionVC by answering the following questions:

- Does our PAICM outperform the state-of-the-art methods?
- What is the effect of NMF in the prototype-guided attribute manipulation?
- How does the proposed PAICM perform in the complementary fashion item retrieval?

In this section, we first detail the experimental settings and then illustrate the experimental results with the analyses on each above research question.
Table 3: Performance comparison among different approaches in terms of AUC.

<table>
<thead>
<tr>
<th>Approach</th>
<th>AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>POP</td>
<td>0.4206</td>
</tr>
<tr>
<td>RAND</td>
<td>0.5094</td>
</tr>
<tr>
<td>Bi-LSTM</td>
<td>0.5502</td>
</tr>
<tr>
<td>BPR-DAE</td>
<td>0.6026</td>
</tr>
<tr>
<td>ExIBR</td>
<td>0.6366</td>
</tr>
<tr>
<td>PAICM</td>
<td>0.7130</td>
</tr>
</tbody>
</table>

4.1 Experimental Settings

Dataset. To evaluate our PAICM, we adopted the public real-world dataset FashionVC [35] consisting of 20,726 outfits with 14,871 tops and 13,663 bottoms, composed by fashion experts. Each fashion item is associated with a visual image, relevant categories and the title description. In addition, to train the attribute classification networks and obtain the semantic attribute representations of fashion items, we utilized an auxiliary benchmark dataset of DeepFashion [27], comprising 33,881 fashion items, each of which is labeled by 18 attributes with 303 attribute elements. Table 1 shows several attribute examples and the corresponding attribute elements. Due to the uneven distribution of the data, we implemented the data augmentation for certain attribute classes with limited samples by multiple operations (e.g., copy, rotation and shift) with an integrated tool of Keras.

Attribute Representation Learning. Regarding the semantic attribute representation learning, we adopted the architecture similar to AlexNet [16] that consists of 5 convolutional layers followed by 3 fully-connected layers. We randomly divided the auxiliary dataset into two chunks: training set (80%) and testing set (20%), and chose the widely-used cross-entropy loss to train all the other baselines, indicating the superiority of introducing the compatibility modeling. We adopted the architecture of Bi-LSTM in [9] which explores the outfit compatibility by sequentially predicting the next item conditioned on previous ones. In our context, we adapted Bi-LSTM to deal with an outfit comprising of a top and a bottom.

4.2 On Comparison of Approaches (RQ1)

As for the compatibility modeling, we chose the following content-based baselines to evaluate the proposed model.

- **POP**: We used the “popularity” of bottom \( b_j \) to measure its compatibility with top \( i \). Here the “popularity” is defined as the number of tops that has been paired with \( b_j \) in the training set.
- **RAND**: We randomly assigned the compatibility scores of \( s_{ij} \) and \( s_{ik} \) between items.
- **Bi-LSTM**: We chose the bidirectional LSTM model in [9] which explores the outfit compatibility by sequentially predicting the next item conditioned on previous ones. In our context, we adapted Bi-LSTM to deal with an outfit comprising of a top and a bottom.
- **ExIBR**: We extended the image-based recommendation (IBR) method proposed in [28] to ExIBR to handle both the visual data and the structured category label of fashion items.
- **BPR-DAE**: We selected the content-based neural scheme introduced by [35] to jointly model the coherent relation between different modalities of fashion items and the implicit preference among items via a dual autoencoder network.

To compare all the approaches fairly, we utilized both the visual image and category metadata in Bi-LSTM, ExIBR, BPR-DAE and PAICM. Table 3 shows the performance comparison among different approaches. As we can see, PAICM outperforms all the other baselines, indicating the superiority of introducing the semantic attribute representations to the compatibility modeling. One possible explanation is that the compatibility modeling task is indeed to model the complicated interactions among various attributes of fashion items, and our semantic attribute representation seems to be just task-oriented.

Moreover, as the prototype learning plays a pivotal role in our PAICM, we particularly investigate the impact of the number of the prototypes learned by the NMF on the performance of compatibility modeling. For simplicity, we adopted the same number of the compatible and incompatible prototypes, and varied that from 10 to 100 with a step of 10. Figure 4 shows the performance of our PAICM with different numbers of prototypes. We found that...
we utilized the K-means algorithm to divide our positive top-bottom pairs into $L_p$ clusters, and the center of each cluster is treated as the learned compatible prototype. Then according to Eqn. (10) and (16), we can find the discordant attribute and replace it with the corresponding attribute representation of the most similar compatible prototype to obtain the manipulated semantic attribute representation. As our compatibility modeling scheme PAICM is able to measure the compatibility between fashion items, here we adopted the rate of the manipulated pairs with improved compatibility as the evaluation metric. Formally, the rate is defined as $\frac{|M|}{|N|}$, where $N$ denotes the set of negative top-bottom pairs determined by our PAICM model and $M$ refers to the set of negative pairs, whose compatibility get improved by the attribute manipulation.

Figure 6 illustrates the performance comparison between NMF and K-means with different numbers of compatible prototypes. As can be seen, NMF consistently surpasses K-means in all configurations, demonstrating the superiority of NMF in discovering the latent prototypes. Moreover, we found that when the number of the compatible prototype is 60, we can achieve the optimal performance.

4.3 On Prototype-guided Attribute Manipulation (RQ2)

To quantitatively evaluate the effects of NMF in the prototype learning, we compared NMF with K-means [10], the most commonly used unsupervised clustering method [5] that is able to group samples sharing the common characteristics together. In particular, we utilized the K-means algorithm to divide our positive top-bottom pairs into $L_p$ clusters, and the center of each cluster is treated as the learned compatible prototype. Then according to Eqn. (10) and (16), we can find the discordant attribute and replace it with the corresponding attribute representation of the most similar compatible prototype to obtain the manipulated semantic attribute representation. As our compatibility modeling scheme PAICM is able to measure the compatibility between fashion items, here we adopted the rate of the manipulated pairs with improved compatibility as the evaluation metric. Formally, the rate is defined as $\frac{|M|}{|N|}$, where $N$ denotes the set of negative top-bottom pairs determined by our PAICM model and $M$ refers to the set of negative pairs, whose compatibility get improved by the attribute manipulation.

Figure 6 illustrates the performance comparison between NMF and K-means with different numbers of compatible prototypes. As can be seen, NMF consistently surpasses K-means in all configurations, demonstrating the superiority of NMF in discovering the latent prototypes. Moreover, we found that when the number of the compatible prototype is 60, we can achieve the optimal performance.
To assess the practical value of PAICM, we conducted experiments on the complementary fashion item retrieval. Considering the fact that it is time-consuming to rank all the bottoms for each top, we utilized the same strategy in [11] to feed each top $t_i$ appeared in $S_{test}$ as a query, and randomly selected $7$ bottoms as the ranking candidates with only one positive bottom. We fed the candidates into the trained model to acquire their latent representations and calculated the compatibility score $s_{ij}$ according to Eqn. (3), based on which we generated a ranking list of the bottoms for the given top. In this work, we focused on the average position of the positive bottom in the ranking list and thus adopted the mean reciprocal rank (MRR) metric $[13, 43, 44]$. 

In total, there are 1,954 unique tops in the testing set. Due to the sparsity of the real-world dataset, 1,262 (64.59%) tops never appear in $S_{train}$. To comprehensively evaluate the proposed model, we divided tops in the testing set into two ground: observed testing tops and unobserved ones. As shown in Figure 8, PAICM shows superiority over all the other baselines at different numbers of bottom candidates in both scenarios, indicating the robustness and effectiveness of PAICM in complementary clothing item retrieval.

5 CONCLUSION AND FUTURE WORK

In this work, we present a prototype-guided interpretable compatibility modeling scheme, PAICM, which is capable of not only determining the outfit compatibility, but also locating the discordant attribute according to the learned prototype, but ignore the factor of users’ personal preferences in clothing matching. Therefore, in the future, we plan to explore the potential of the user context in complementary clothing matching and attribute suggesting.

ACKNOWLEDGMENTS

This work is supported by the National Natural Science Foundation of China, No.: 61702300, No.: 61772310, No.: 61702302, No.: 61802231, and No.: U1836216; the Project of Thousand Youth Talents 2016; the Tencent AI Lab Rhino-Bird Joint Research Program, No.: JR201805; the Future Talents Research Funds of Shandong University, No.: 2018WLJH63.