Complementary Factorization towards Outfit Compatibility Modeling

Tianyu Su
Shandong University
tyanyu.su@gmail.com

Xuemeng Song*
Shandong University
sxmustc@gmail.com

Na Zheng
Shandong University
zhengnagrape@gmail.com

Weili Guan
Monash University
weili.guan@monash.edu

Yan Li
Kuaishou Technology
liyan@kuaishou.com

Liqiang Nie*
Shandong University
nieliqiang@gmail.com

ABSTRACT

Recently, outfit compatibility modeling, which aims to evaluate the compatibility of a given outfit that comprises a set of fashion items, has gained growing research attention. Although existing studies have achieved prominent progress, most of them overlook the essential global outfit representation learning, and the hidden complementary factors behind the outfit compatibility uncovering. Towards this end, we propose an Outfit Compatibility Modeling scheme via Complementary Factorization, termed as OCM-CF. In particular, OCM-CF consists of two key components: *context-aware outfit representation modeling* and *hidden complementary factors modeling*. The former works on adaptively learning the global outfit representation with graph convolutional networks and the multi-head attention mechanism, where the item context is fully explored. The latter targets at uncovering the latent complementary factors with multiple parallel networks, each of which corresponds to a factor-oriented context-aware outfit representation modeling. In this part, a new orthogonality-based complementarity regularization is proposed to encourage the learned factors to complement each other and better characterize the outfit compatibility. Finally, the outfit compatibility is obtained by summing all the hidden complementary factor-oriented outfit compatibility scores, each of which is derived from the corresponding outfit representation. Extensive experiments on two real-world datasets demonstrate the superiority of our OCM-CF over the state-of-the-art methods.

CCS CONCEPTS

• **Information systems → Retrieval tasks and goals**; World Wide Web.

KEYWORDS

Fashion Analysis; Complementary Compatibility Modeling; Representation Learning

ACM Reference Format:
Tianyu Su, Xuemeng Song, Na Zheng, Weili Guan, Yan Li, and Liqiang Nie. 2021. Complementary Factorization towards Outfit Compatibility Modeling. In *Proceedings of the 29th ACM International Conference on Multimedia (MM ’21), October 20–24, 2021, Virtual Event, China.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3474085.3475537

1 INTRODUCTION

With the prosperity of e-commerce fashion market, online shopping platforms (e.g., Amazon[1], eBay[2], and Taobao[3]) need more intelligent systems to satisfy the growing demand of customers, where the deep learning [9, 10, 19] technique is explored. In particular, as a fundamental tool for various downstream industrial applications, such as personalized-fashion design [11, 42], outfit recommendation [14, 16, 20, 25], and fashion-oriented dialogue system [17, 24], the automatic fashion compatibility modeling, which aims to estimate whether the given set of fashion items makes a compatible outfit, has attracted increasing research attention.

Existing studies for the fashion compatibility modeling can be broadly split into three groups, i.e. pair-based [18, 31, 32, 34, 38], sequence-based [6, 23], and graph-based methods [1, 2, 14, 40]. To be specific, the pair-based methods model the outfit compatibility by separately estimating the compatibility of each pair of items in an outfit. Apparently, the pair-based methods overlook the hyper-relation among multiple fashion items, and hence only reach the suboptimal performance. Beyond that, the sequence-based and graph-based methods directly evaluate the outfit compatibility by treating the outfit as a whole. Specifically, sequence-based methods treat an outfit as a fixed order of items, and graph-based methods regard the outfit as an item graph.

Although these efforts have achieved promising results, they mainly have two limitations: 1) They decouple the outfit compatibility into either the compatibility of individual item toward the outfit, or of the pair of items, and thus focus on learning the representation of each composing item. We argue that this manner still fails to authentically treat the outfit as a whole, namely, it overlooks the global outfit representation learning. And 2) they evaluate the outfit compatibility based on the single latent compatibility space, while we argue that the outfit compatibility can be measured in multiple hidden spaces, since it is essentially affected by multiple

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

MM ’21, October 20–24, 2021, Virtual Event, China

ACM ISBN 978-1-4503-8651-7/21/10…$15.00
https://doi.org/10.1145/3474085.3475537

[1] https://www.amazon.com/.
[2] https://www.ebay.com/.
[3] https://www.taobao.com/.

complementary hidden factors, like the color, style, shape, and material. Accordingly, to address the above research limitations, in this work, we aim to estimate the compatibility of the outfit from multiple factors with the global outfit representation learning.

However, this is a non-trivial task due to the following challenges. 1) The key of the outfit compatibility modeling is to learn the global outfit representation that encodes the outfit's compatibility. As the global outfit representation cannot be discussed without the local item representation learning, how to derive the accurate item representation that compiles its compatibility to all the other items poses the first challenge for us. 2) Since each outfit involves a variable number of composing items, and different items contribute to the outfit differently, how to adaptively learn the global outfit representation based on the item representation is a crucial challenge. And 3) in a sense, the hidden factors complementarily characterize the outfit compatibility, such as the color-oriented, material-oriented, and style-oriented compatibility. Therefore, how to model the complementarity of these hidden factors and boost the outfit compatibility modeling constitutes another tough challenge.

To tackle these challenges, we devise a novel outfit compatibility modeling scheme, termed as OCM-CF. As shown in Figure 1, OCM-CF contains two essential components: *context-aware outfit representation modeling* and *hidden complementary factors modeling*. Specifically, the context-aware outfit representation modeling focuses on learning the global representation of the outfit. In particular, we adopt graph convolutional networks (GCNs) to flexibly support the compatibility modeling for the outfit with an arbitrary number of fashion items. During the information propagation, different from existing studies that only propagate the item embedding, we focus on propagating the item-item relationship, and propose an adaptive item-item relationship propagation module based on the gate mechanism. In addition, to derive the global outfit representation, we employ the multi-head attention mechanism to encourage the global outfit representation to fully incorporate the context information of each fashion item. Pertaining to the hidden complementary factors modeling, we introduce a few parallel branches, each of which is deployed with the network of the first component, i.e., context-aware outfit representation modeling, and works on exploring the outfit compatibility on one exclusive complementary hidden factor. To encourage each branch to concentrate on learning one aspect and making the whole scheme indeed comprehensive, we introduce the orthogonality-based complementarity regularization to avoid the factor homogenization.

Our main contributions can be summarized in threefold:

- To our best knowledge, we are the first attempt to fulfill the outfit compatibility modeling by directly learning the context-aware global outfit representation.
- We propose an orthogonality-based complementarity regularization to promote the outfit compatibility estimation from multiple complementary hidden factors.
- We conduct extensive experiments on two real-world datasets, and the results show the superiority of OCM-CF over the state-of-the-art methods. As a byproduct, we have released the codes to benefit other researchers[4].

The rest of the paper is organized as follows. Section 2 briefly reviews the related work. And then Section 3 details the proposed OCM-CF. The experimental results and analyses are introduced in Section 4, followed by the conclusion and future work in Section 5.

2 RELATED WORK

2.1 Fashion Compatibility Modeling

Recently, increasing research interest has been drawn to the domain of fashion compatibility modeling. Initially, researchers mainly focused on the compatibility modeling between two fashion items. For instance, Song et al. [29] devised a multi-modal fashion compatibility approach for two fashion items based on neural networks. Later, considering that the outfit usually involves multiple items in real-world applications, an amount of researches [1, 2, 6, 14, 31, 32] have been dedicated to studying the outfit compatibility modeling.

Existing methods on the outfit compatibility modeling can be roughly divided into three groups. The first group comprises the pair-based methods [30, 35] which focus on learning the outfit compatibility based on evaluating the compatibility of each item pair in the outfit. Apparently, the pair-based methods overlook the context relationship among items and they are computationally inefficient since the time complexity is $O(N^2)$ for an outfit composing N fashion items. The second group [6, 23] consists of the sequence-based methods, which treat the outfit as a sequence of fashion items in a fixed predefined order. Although they take into account the item context, their performance tends to be sensitive to the order of items [2, 41], and it is non-trivial to predefine the item order in the outfit reasonably. The third group contains the graph-based methods [1, 2, 14], which introduce graph neural networks (GNNs) [4] into the outfit compatibility modeling. For example, Cui et al. [2] and Li et al. [14] resorted to build the item graph for each outfit, and employed GCNs to fulfill the outfit compatibility modeling task. Although these efforts have achieved compelling success, they neglect the importance of the global representation of the outfit in characterizing the complex compatible relationships among items, and evaluate the outfit compatibility with a general compatibility space. Beyond that, we explored the global outfit representation learning with hidden complementary factors uncovering.

2.2 Graph Neural Network

Due to the remarkable capability of dealing with the unstructured data, like a graph, GNNs have been adopted in many research domains, such as the node classification [13, 27], image retrieval [44, 45], and personalized recommendation [8, 39]. Initially, Gori et al. [4] proposed GNNs to model a set of items and their relationship. Later, GCNs [12, 13] are devised to introduce the convolution operation into the graph domain by updating each node's representation via aggregating information from its neighbor nodes. In order to improve the model generalization ability, Velickovic et al. [36] devised a graph attention network, which assigns different importance to different neighbor nodes during the graph propagation, while Hamilton et al. [5] proposed a general inductive framework that can leverage node features to efficiently generate node embeddings for unseen data by learning aggregator functions. Inspired by the success from these studies, in this work, we employed GCNs to support the compatibility modeling for the outfit with a variable

[4] https://aoecode.wixsite.com/ocm-cf/.

length, where we developed an adaptive item-item relationship propagation module based on the gate mechanism to promote the outfit compatibility modeling performance.

3 METHODOLOGY

In this section, we first formally define the research task and then detail the proposed OCM-CF.

3.1 Problem Formulation

Formally, suppose we have a set of positive (well-composed) outfits $\mathcal{S} = \{s^1, s^2, \cdots, s^T\}$ and a set of fashion items $\mathcal{X}$. Each outfit is associated with a set of m fashion items, denoted as $s = \{x_1, x_2, \cdots, x_m\}$, where x_j is the j-th item of the outfit. Notably, m is a variable, which differs for different outfits. Each item x_j has a product image denoted as I_j and a category metadata denoted as $C_v \in \mathcal{C}$, $v \in \{1, 2, \cdots, N_c\}$, where $\mathcal{C} = \{C_1, C_2, \cdots, C_{N_c}\}$ refers to the whole set of N_c categories used for organizing all the fashion items.

In this work, we aim to devise an outfit compatibility modeling network $\mathcal{F}$, which is capable of assessing the overall compatibility score of a given outfit s as follows,

$$\hat{y} = \mathcal{F}(\{x_j\}_{j=1}^{m} | \Theta_F), \tag{1}$$

where $\hat{y}$ denotes the estimated compatibility score of the given outfit and Θ_F is a set of to-be-learned model parameters.

3.2 Context-aware Outfit Representation Learning (CORL)

We argue that the essence of the outfit compatibility modeling is to learn a precise outfit representation that captures the compatibility among all its composition items. Due to the remarkable performance of GCNs in unstructured data representation learning, we employ GCNs to handle the outfit representation learning.

Item Visual Embedding. To begin, we first extract the image feature via the Convolutional Neural Network (CNN) model, which can be defined as follows,

$$\mathbf{f}_j = \mathbf{CNN}(I_j; \Theta_{cnn}), \tag{2}$$

where $\mathbf{f}_j \in \mathbb{R}^d$ denotes the image embedding of the item x_j, d is the embedding size, and Θ_{cnn} refers to the parameters of the **CNN** model. Concretely, following [31, 32], we adopt a 18-layer Deep Residual Network [7] pretrained on ImageNet [28]. To alleviate the overfitting, we use the L$_2$ regularization on the learned image embedding [32, 34], as follows,

$$\mathcal{L}_2(s) = \sum_{j=1}^{m} \|\mathbf{f}_j\|_2. \tag{3}$$

Outfit Graph Construction. Formally, the graph for the outfit s can be defined as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{v_1, v_2, \cdots, v_m\}$ refers to the set of item nodes, while $\mathcal{E} = \{(v_i, v_j, e_{ij}) | i \neq j, v_i \in \mathcal{V}, v_j \in \mathcal{V}\}$ stands for the set of edges linking these item nodes. The triplet (v_i, v_j, e_{ij}) denotes the edge from the node v_i to node v_j weighted by e_{ij}. Regarding the node representation initialization, since the visual cue is essential for the compatibility reasoning, we initialize each node embedding, denoted as $\mathbf{v}_j^0, j = 1, 2, \cdots, m$, with the corresponding item's visual embedding, i.e., $\mathbf{v}_j^0 = \mathbf{f}_j$. Pertaining to the edge weight, instead of setting all the edge weights as the constant, we resort to the category co-occurrence probability, due to the concern that an item should pay more attention to those items whose categories frequently co-occurred with its own category to attentively absorb the neighbors' information. For example, according to the category occurrence derived from our dataset, a T-shirt should attend the pant more as compared to the pair of glasses in the same outfit.

Towards this end, we introduce the category correlation matrix $\mathbf{M} \in \mathbb{R}^{N_c \times N_c}$ in a data-driven manner, which is defined as follows,

$$\begin{cases} P(C_u|C_v) = \dfrac{n_1(C_u, C_v)}{n_2(C_v)}, \\ \mathbf{M}_{uv} = \dfrac{P(C_u|C_v)}{\sum_{k=1}^{N_c} P(C_u|C_k)}, \end{cases} \tag{4}$$

where $P(C_u|C_v)$ denotes the occurrence probability of category C_u given category C_v. $n_1(C_u, C_v)$ is the function for counting the concurrence times of categories C_u and C_v in the training dataset, and $n_2(C_v)$ is that for counting the occurrence times of category C_v in the training dataset. Suppose that items x_i and x_j belong to the categories C_u and C_v, respectively. Then we define the weight for the edge from x_i to x_j as,

$$e_{ij} = \mathbf{M}_{uv}. \tag{5}$$

Item-Item Relationship Propagation (IRP). Different from existing work [1, 2] that propagates the pure neighbor items' embedding over the item graph, we propose to propagate the item-item relationship embedding, a.k.a., Adaptive Relationship Derivation, which plays the pivotal role in the outfit compatibility modeling. Meanwhile, we argue that the high-order connectivities are beneficial to synthesize a richer node representation [8, 39], and thus stack L propagation layers to exploit the item-item relationship. Specifically, we define the relationship embedding between the item i and j regarding the l-th propagation layer as $\mathbf{q}_{ij}^l = \mathbf{v}_i^l \otimes \mathbf{v}_j^l$, where $\otimes$ denotes the element-wise product operation, and $l = 1, 2, \cdots, L$.

Regarding the item-item relationship propagation, we argue that different dimensions of the relationship embedding may contribute differently to the compatibility modeling. Accordingly, we introduce the gate mechanism to adaptively propagate the item-item relationship. In particular, the gate function is defined as follows,

$$\mathbf{r}_{ij}^l = \sigma\Big(\mathbf{W}_1^l \delta\Big(\mathbf{W}_2^l (\mathbf{v}_i^l || \mathbf{v}_j^l) + \mathbf{b}_2^l\Big) + \mathbf{b}_1^l\Big), \tag{6}$$

where $\mathbf{r}_{ij}^l \in \mathbb{R}^d$ is the gate mask for the item pair (x_i, x_j) in the l-th propagation layer, $||$ is the concatenation operation, $\sigma(\cdot)$ and $\delta(\cdot)$ are Sigmoid and LeakyReLU [21] activate functions, respectively. $\mathbf{W}_1^l \in \mathbb{R}^{d \times d}$, $\mathbf{W}_2^l \in \mathbb{R}^{d \times 2d}$, $\mathbf{b}_1^l \in \mathbb{R}^d$, and $\mathbf{b}_2^l \in \mathbb{R}^d$ are trainable parameters of the fully connected layers for the l-th order relationship propagation. Based upon the gate function, we formulate the final item-item relationship as $\mathbf{g}_{ij}^l = \delta(\mathbf{r}_{ij}^l \otimes \mathbf{q}_{ij}^l)$.

Thereafter, we aggregate all the neighbor relationships to refine the ego item representation. Mathematically, the item-item relationship propagation for item j in the l-th order propagation can be formulated as,

$$\mathbf{v}_j^{(l+1)} = \delta\Big(\mathbf{W}_3^l \Big(\mathbf{f}_j^l + \sum_{i \in N_j} e_{ij} \mathbf{g}_{ij}^l\Big) + \mathbf{b}_3^l\Big), \tag{7}$$

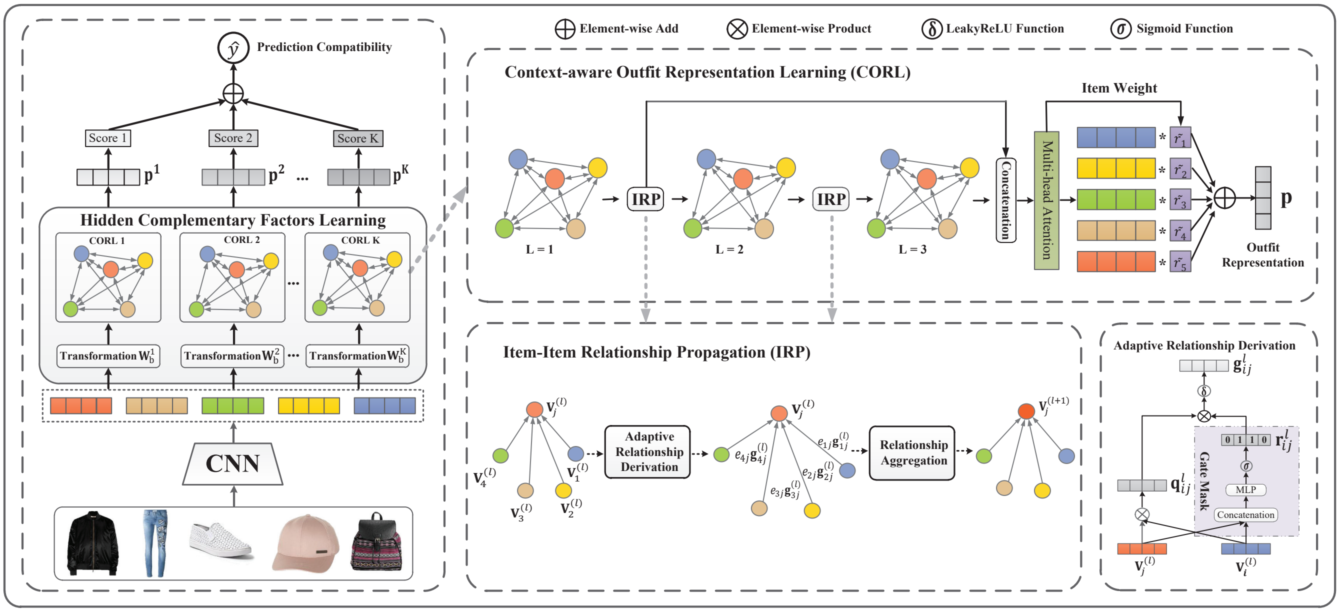


Figure 1: Illustration of the proposed OCM-CF. Left: the overall scheme that employs a set of K parallel branches for the hidden complementary factors learning, where each branch corresponds to a factor-oriented context-aware outfit representation learning (CORL). Right: the detailed CORL component, and its adaptive item-item relationship propagation module.

where $\mathcal{N}_j$ is the set of neighbor nodes of the node x_j. $\mathbf{W}_3^l$ and $\mathbf{b}_3^l$ are learnable parameters for the node information aggregation in the l-th propagation layer. Finally, to avoid the information loss during the item-item relationship propagation, we incorporate the initial visual embedding to define the final item embedding as follows,

$$\hat{\mathbf{v}}_j = \mathbf{v}_j^0 \| \mathbf{v}_j^L, \tag{8}$$

where $\hat{\mathbf{v}}_j \in \mathbb{R}^{2d}$ is the final representation of the item x_j.

To encourage the gate mask to filter the discriminative dimensions of the relationship, we introduce the L1 regularization to enhance the sparsity of gate masks as follows,

$$\mathcal{L}_1(s) = \sum_{l=1}^{L} \sum_{i=1}^{m} \sum_{j=1, j \neq i}^{m} \|\mathbf{r}_{ij}^l\|_1. \tag{9}$$

Global Outfit Representation. Different from existing graph-based compatibility modeling methods [2, 14] that focus on learning the individual compatibility of each item toward the outfit based on the local item representation learning, we directly target at the global outfit representation learning. As to discriminate the importance of different items in characterizing the outfit, we adopt the multi-head self-attention mechanism [33] to summarize the outfit representation from the set of item representations. For simplicity, we pack all item embeddings into a matrix $\widehat{\mathbf{V}}^F = [\hat{\mathbf{v}}_1; \hat{\mathbf{v}}_2; \cdots; \hat{\mathbf{v}}_m] \in \mathbb{R}^{m \times 2d}$. Suppose we have h attention heads, and the self-attention function of the i-th attention head can be formulated as follows,

$$\begin{cases} \mathbf{S}_i^a = softmax(\frac{\mathbf{Q}_i \mathbf{K}_i^{\mathrm{T}}}{\sqrt{d_k}}), \\ \mathbf{H}_i = \mathbf{S}_i^a \mathbf{Z}_\mathbf{i}, \end{cases} \tag{10}$$

where $\mathbf{Q}_i = \widehat{\mathbf{V}}^F \mathbf{W}_i^Q$, $\mathbf{K}_i = \widehat{\mathbf{V}}^F \mathbf{W}_i^K$, and $\mathbf{Z}_i = \widehat{\mathbf{V}}^F \mathbf{W}_i^Z$ refer to the query, key and value matrices, respectively, while $\mathbf{W}_i^Q \in \mathbb{R}^{2d \times d_q}$, $\mathbf{W}_i^K \in \mathbb{R}^{2d \times d_k}$, and $\mathbf{W}_i^Z \in \mathbb{R}^{2d \times d_z}$ are the corresponding trainable linear projection matrices. d_q, d_k, and d_z are the dimensions of the latent space, where $d_q = d_k = d_z = \frac{2d}{h}$. The $softmax$ operation is performed for each row. $\mathbf{S}_i^a \in \mathbb{R}^{m \times m}$ denotes the attention weight matrix of the i-th head, the (j, k)-th entity of which reflects the importance of the k-th item to the j-th item. $\mathbf{H}_i \in \mathbb{R}^{m \times d_z}$ is the output of i-th head, with each row referring to an item local feature.

Based upon the attention weight matrices derived from the h heads, we define the importance of the i-th item as the summation of its importance to all the items in the outfit. Formally, we have,

$$\begin{cases} r_i = \sum_{j=1}^{m} \bar{\mathbf{S}}(j, i), i = 1, 2, \cdots, m, \\ \tilde{r}_i = \frac{\exp(r_i)}{\sum_{k=1}^{m} \exp(r_k)}, \end{cases} \tag{11}$$

where $\bar{\mathbf{S}} = \frac{1}{h} \sum_{i=1}^{h} \mathbf{S}_i^a$, and $\tilde{r}_i$ is the normalized importance of the item i to characterize the outfit. Finally, we derive the outfit representation as follows,

$$\mathbf{p} = \sum_{i=1}^{m} \tilde{r}_i \mathbf{H}^F(i, :), \tag{12}$$

where $\mathbf{H}^F = [\mathbf{H}_1, \mathbf{H}_2, \cdots, \mathbf{H}_h] \in \mathbb{R}^{m \times 2d}$, and $\mathbf{H}^F(i, :)$ is the i-th row of $\mathbf{H}^F$, representing the i-th item representation. In the end, our proposed context-aware outfit representation learning network can be summarized as,

$$\mathcal{F}_{out}(\mathcal{G}) = \mathbf{p}, \tag{13}$$

where $\mathcal{G}$ is the item graph of outfit s.

3.3 Hidden Complementary Factors Learning

As forementioned, each outfit compatibility can be affected by multiple complementary hidden factors, like the color, style, shape, and material. Accordingly, in this part, we propose the hidden complementary factors learning method. In particular, we first project each outfit into multiple complementary factor subspaces, in which the factor-oriented compatibility can be modeled.

To this end, for each outfit s, we introduce K parallel branches, denoted as $\mathcal{B}_1, \cdots, \mathcal{B}_K$, where each branch comprises a network for CORL, and focuses on one hidden factor-oriented outfit compatibility reasoning. Specifically, each branch $\mathcal{B}_k$, $k = 1, 2, \cdots, K$ can be formulated as,

$$\mathbf{p}^k = \mathcal{F}_{out}^k(\mathcal{G}^k), \tag{14}$$

where $\mathbf{p}^k \in \mathbb{R}^{2d}$ denotes the global outfit representation pertaining to the k-th hidden factor. $\mathcal{G}^k = \{\mathcal{V}^k, \mathcal{E}^k\}$ is the outfit graph designed for the k-th branch, where $\mathcal{E}^k = \mathcal{E}$, namely, all the branches share the same graph structure. To facilitate the hidden factor learning, we initialize the node representations of different branches with different hidden representations. In particular, we have,

$$\mathbf{f}_j^k = \mathbf{W}_b^k \mathbf{f}_j, k = 1, 2, \cdots, K, \tag{15}$$

where $\mathbf{f}_j$ is the initial visual embedding of item j, defined in Eqn. (2). $\mathbf{W}_b^k \in \mathbb{R}^{d \times d}$ is the weight matrix for transforming the visual embedding of each item into the k-th hidden factor space. $\mathbf{f}_j^k \in \mathbb{R}^d$ denotes the k-th hidden factor-oriented item representation.

It is worth noting that, with no constraint, the learned hidden factors tend to present the homogenization, resulting in redundant compatibility reasoning of different branches [37]. To encourage different branches to model different hidden factors, we thus introduce the orthogonality-based complementarity regularization. Formally, we have the following objective function,

$$\mathcal{L}_{com}(s) = \sum_{j=1}^{m} ||\mathbf{F}_j \mathbf{F}_j^{\mathrm{T}} - \mathbf{I}||_F^2, \tag{16}$$

where $\mathbf{I} \in \mathbb{R}^{K \times K}$ is the identity matrix. $\mathbf{F}_j = [\mathbf{f}_j^1; \mathbf{f}_j^2; \cdots; \mathbf{f}_j^K] \in \mathbb{R}^{K \times d}$ denotes different factor embeddings of the fashion item j, and $|| \cdot ||_F$ denotes the Frobenius norm of matrix.

3.4 Outfit Compatibility Modeling

Based on the hidden factor-oriented outfit representations, i.e., $\mathbf{p}^1, \mathbf{p}^2, \cdots, \mathbf{p}^K$, we employ a linear transformation to obtain the compatibility score $\hat{y}$ for the given outfit s as follows,

$$\hat{y} = \sum_{k=1}^{K} \mathbf{W}_s^k \mathbf{p}^k, \tag{17}$$

where $\mathbf{W}_s^k \in \mathbb{R}^{1 \times 2d}$ is the weight matrix for the branch $\mathcal{B}_k$.

Similar to existing methods [2, 30], to exploit the implicit compatibility preference among fashion items, we also adopt the Bayesian Personalized Ranking (BPR) [26] loss, which encourages the score of the positive outfit higher than that of the negative one. Accordingly, we first build the following training set $\mathcal{D} = \{(s^+, s^-)\}$, where s^+ and s^- denote the positive and negative outfit samples, respectively. s^+ is directly sampled from the positive outfit set $\mathcal{S}$, while s^- is strategically sampled. The sampling details will be given in the experiment section. For each training pair (s^+, s^-), we have the following objective function,

$$\mathcal{L}_{bpr}(s^+, s^-) = -\ln \sigma(\hat{y}_{s^+} - \hat{y}_{s^-}). \tag{18}$$

Then the ultimate training loss can be defined as follows,

$$\begin{aligned} \min_{\Theta_F} \mathcal{L} = \sum_{(s^+, s^-) \in \mathcal{D}} & \mathcal{L}_{bpr}(s^+, s^-) + \lambda_1 \big(\mathcal{L}_{com}(s^+) + \mathcal{L}_{com}(s^-)\big) \\ & + \lambda_2 \big(\mathcal{L}_1(s^+) + \mathcal{L}_1(s^-)\big) + \lambda_3 \big(\mathcal{L}_2(s^+) + \mathcal{L}_2(s^-)\big), \end{aligned} \tag{19}$$

where λ_1, λ_2, and λ_3 are non-negative trade-off hyper-parameters and Θ_F refers to the set of parameters (i.e., Θ_{cnn}, $\mathbf{W}_1^l$, $\mathbf{W}_2^l$, $\mathbf{W}_3^l$, $\mathbf{b}_1^l$, $\mathbf{b}_2^l$, $\mathbf{b}_3^l$, $\mathbf{W}_i^Q$, $\mathbf{W}_i^K$, $\mathbf{W}_i^Z$, $\mathbf{W}_b^k$ and $\mathbf{W}_s^k$) of the model.

4 EXPERIMENTS

To evaluate the proposed method, we conducted extensive experiments on the two real-world datasets Polyvore Outfits and Polyvore Outfits-D via answering the following research questions:

- **RQ1:** Does OCM-CF surpass the state-of-the-art methods?
- **RQ2:** How does each component affect our OCM-CF?
- **RQ3:** What is the qualitative performance of OCM-CF?

4.1 Dataset

In this work, we adopted the Polyvore dataset [32] with two versions: Polyvore Outfits and Polyvore Outfits-D. The difference between these two versions lies in that the former has overlapping items between its training and testing sets, while the latter does not. Polyvore Outfits contains 68, 306 outfits composed by 365, 054 fashion items, where the average number of items in an outfit is 5.3. Polyvore Outfits-D is relatively smaller than Polyvore Outfits, which consists of 32, 140 outfits composed by 175, 485 fashion items, where the average size of outfits is 5.1.

4.2 Implementation Details

Negative Outfit Composition. Regarding the training dataset construction, we set the ratio of positive and negative samples to 1 : 1. Considering that human's cognitive learning is an easy-to-hard process, analogically, we made the model to first learn from the easy cases, and hence adopted the following three manners to compose a negative outfit s^- for each positive outfit s^+: 1) Manner1: randomly sample $|s^+|$ items from $\mathcal{X}$ without any restriction; 2) Manner2: randomly sample $|s^+|$ items from $\mathcal{X}$ according to the item categories of s^+; and 3) Manner3: randomly choose one item of the positive outfit and replace it with a randomly sampled item of the same category. Intuitively, in the first few epochs, we used Manner1 to derive the negative samples, then Manner2, followed by Manner3 in the last few epochs.

Experiment Setting. In Polyvore dataset, each fashion item is assigned with both the coarse-grained category, like *Top*, and the fine-grained category, like *T-shirt*. Due to the concern of the highly imbalanced data distribution with hundreds of fine-grained categories, which may degrade the model generalization performance, we resorted to the coarse-grained category metadata to derive the edge weight between each two items. The Adam optimizer is employed with mini-batch size 64 and embedding size $d = 64$. The learning rate is set as $5e^{-5}$ with the exponential decay 0.985 of each

Table 1: Performance comparison among different methods on three tasks. Our results are highlighted in bold.

Method	Polyvore Outfits		Polyvore Outfits-D		Polyvore Outfits			Polyvore Outfits-D		
	Compat. AUC	FITB Acc	Compat. AUC	FITB Acc	HR@5	HR@10	HR@40	HR@5	HR@10	HR@40
Bi-LSTM	0.68	42.20%	0.65	40.10%	0.032	0.076	0.244	0.052	0.088	0.249
SCE-NET	0.83	52.80%	0.82	52.10%	0.079	0.143	0.340	0.076	0.129	0.334
Type-aware	0.87	56.60%	0.78	47.30%	0.108	0.165	0.372	0.040	0.072	0.236
NGNN	0.75	53.02%	0.68	42.49%	0.084	0.136	0.341	0.033	0.068	0.219
Context-aware	0.81	55.63%	0.77	50.34%	0.106	0.163	0.384	0.083	0.132	0.325
HFGN	0.84	49.90%	0.70	39.03%	0.050	0.080	0.288	0.023	0.049	0.164
OCM-CF	**0.92**	**63.62%**	**0.86**	**56.59%**	**0.145**	**0.238**	**0.502**	**0.096**	**0.158**	**0.370**
%Improv.	5.75%	12.40%	4.88%	8.62%	34.26%	44.24%	30.73%	15.66%	19.70%	10.78%

epoch. We empirically set $L = 3$ as the propagation layers, and we stacked 3 layers of multi-head self-attentions with $h = 8$ heads. In the hidden complementary factors learning module, the number of branch K is set to 5. We set $\lambda_2 = 5e^{-4}$ and other λ parameters in Eqn. (19) to $5e^{-3}$. The proposed model is trained for 80 epochs, and the performance is reported on the test dataset. Notably, during the training, we used two thresholds regarding the epoch number to switch the manner of negative outfit composition as 10 and 40. We only used the image signal in all experiments for fair comparisons.

Evaluation Tasks and Metrics. We evaluated our proposed OCM-CF by conducting experiments on three popular tasks: the outfit compatibility prediction [6, 32], fill-in-the-blank [2, 31], and complementary fashion item retrieval [3, 30]. 1) The task of outfit compatibility prediction is to evaluate the compatibility score of a given outfit that contains an arbitrary number of fashion items. Following existing studies [6, 32], we adopted the AUC (Area Under the ROC curve) [43] as the evaluation metric. 2) The task of fill-in-the-blank (FITB) is to select the most compatible item from a set of item candidates for an incomplete query outfit. To prepare the data, for each positive/compatible outfit, we randomly selected out an item as the target item, and set the rest items of the outfit as the query. Then we composed the target item with three other randomly selected items of the same category with the target item from the dataset as the candidate item choices. To handle the task, we composed each candidate item with query items as an outfit, and used the well-trained model to compute each outfit’s compatibility score. Based on that, we chose the item with the highest score as the answer, and used the accuracy as the evaluation metric. 3) The task of complementary fashion item retrieval can be seen as an extension of the FITB task. Concretely, we extended the size of the candidate item set to 500, where there is only one positive (target) item and 499 negative items of the same category. We adopted the Hit Rate (HR) at 5, 10 and 40 to evaluate the model performance.

4.3 On Model Comparison (RQ1)

To validate the effectiveness of our proposed method, we compared it with the following state-of-the-art methods, including the pair-based, sequence-based, and graph-based models.

- **Bi-LSTM** [6] permutes all items of an outfit into a predefined order according to the item category, and cast the outfit compatibility modeling as a sequence prediction problem, where bidirectional LSTMs are used. For fairness, we removed the text information from the released model.
- **Type-aware** [32] measures the fashion item compatibility with type-respecting spaces rather than a single general space. We used the code provided by authors, and re-trained the model with only the image cue.
- **SCE-NET** [31] learns different similarity conditions and employs a weight module to combine all different embeddings as a fashion item representation. Similar to Type-aware, we removed the regularization of the text information from the author released model.
- **NGNN** [2] maps the fashion item feature into a category space to build the item graph, where the node embedding is updated based on GRU [15], and the attention mechanism is used for summarizing the outfit compatibility score.
- **Context-aware** [1] builds a graph with all fashion items in the dataset. Each node will receive message from its own outfit and other outfits to learn the contextual item embedding. In the testing stage, we computed the compatibility score based on its own embedding.
- **HFGN** [14] different from NGNN, devises a R-view attention map and a R-view score map to assess the outfit compatibility score based on GCNs over the category-oriented outfit graph.

Table 1 shows the performance comparison among different approaches on both Polyvore Outfits and Polyvore Outfits-D datasets under different tasks. For clarity, we divided the baselines into three groups, i.e., sequence-based, pair-based, and graph-based models. From this table, we have the following observations: 1) Compared to other baselines, Bi-LSTM achieves the worst performance on most of evaluation metrics, which may be due to two facts. On the one hand, essentially, it is inappropriate to model the outfit as an ordered list of fashion items. On the other hand, this method computes the outfit compatibility score by predicting the next item with the previous ones, which may cause the cumulative error propagation. 2) Unexpectedly, the graph-based baselines, i.e., NGNN, Context-aware and HFGN, do not show superiority over the pair-based methods, i.e., SCE-NET and Type-aware. The possible explanation for Context-aware is that this method learns fashion item embeddings in a single space, while the pair-based one, Type-aware, considers the visual similarity from different metric spaces. For NGNN and HFGN, they employ fashion item embeddings in a category space to initialize nodes, which leads to the category bias, namely, the model maybe learn compatibility patterns at the category level, resulting in inaccurate evaluation of outfit compatibility score. And 3) our proposed method OCM-CF consistently achieves the best performance on all tasks. It is worth

Table 2: Performance comparison of the ablation study.

Method	Polyvore Outfits		Polyvore Outfits-D	
	Compat. AUC	FITB Accuracy	Compat. AUC	FITB Accuracy
OCM-CF	**0.92**	**63.62%**	**0.86**	**56.59%**
w/o Edge Weight	0.89	62.60%	0.84	55.64%
w/o Relationship	0.90	62.12%	0.84	55.25%
w/o Attention	0.64	51.12%	0.61	34.11%
w Fine-grained	0.87	61.93%	0.80	54.65%
w/o Complementarity	0.89	62.83%	0.80	55.94%

noting that our method has large improvements on the complementary fashion item retrieval task *w.r.t.* HR@5 and HR@10, which is meaningful for the real-world application since users can quickly find the complementary fashion item fitting the outfit. The results verify the superiority of our model over the state-of-the-art methods, and the effectiveness of our contextual outfit representation learning and the hidden complementary factors learning.

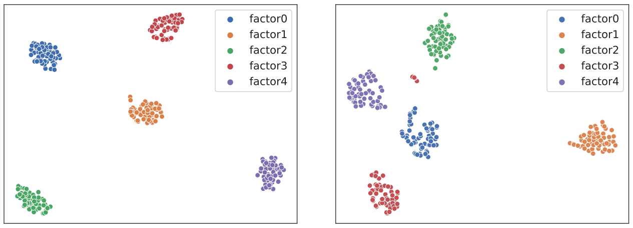
4.4 On Ablation Study (RQ2)

To investigate how each component affects our model, we introduced the following five variants:

- **w/o Edge Weight**. In this variant, we set all the edge weights as a constant 1.
- **w/o Relationship**. We modified the $\mathbf{g}_{ij}^{l} = \mathbf{v}_{i}^{l}$ in Eqn. (7) for only aggregating neighbour item embeddings.
- **w/o Attention**. We replaced the multi-head attention mechanism with a mean pooling operation over the representation of all the composition fashion items.
- **w Fine-grained**. We derived the edge weight with the fine-grained category co-occurrence [2, 14] rather than the coarse-grained category used in our OCM-CF.
- **w/o Complementarity**. We removed the orthogonality-based complementarity regularization from the hidden complementary factors learning.

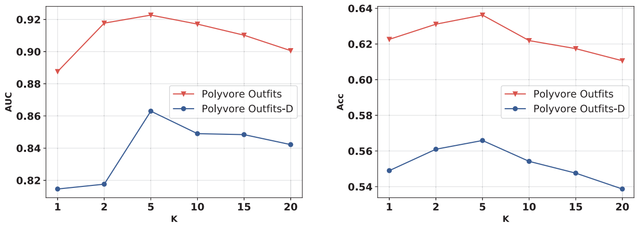
Table 2 shows the performance comparison of different methods in the ablation study. From Table 2, we noticed that all the variants degrade the performance of our OCM-CF, which indicates the importance of each component. In particular, firstly, w/o Edge Weight performs worse than OCM-CF, implying that utilizing the category co-occurrence probability can promote the item-item relationship propagation. Secondly, w Fine-grained is inferior to OCM-CF, which confirms our assertion that utilizing the fine-grained categories may involve the highly imbalanced data distribution, making the co-occurrence pattern unreliable. Thirdly, the inferior performance of w/o Relationship suggests that propagating the item-item relationship is more meaningful for the outfit compatibility modeling. Fourthly, we found the performance of w/o Attention significantly drops, as compared to OCM-CF, demonstrating the necessity of deriving the global outfit representation in an attentive manner. Lastly, w/o Complementarity is also inferior to OCM-CF, reflecting the effectiveness of our proposed complementarity regularization.

To gain deeper insights regarding the complementarity regularization, we visualized the factor-oriented representations for the outfits randomly sampled from the test dataset obtained by our model and its variant w/o Complementarity with the tool of t-SNE [22] in Figure 2. We observed that the distance between clusters derived by OCM-CF is larger than that by w/o Complementarity, which reflects the outfit representations for different factors learned by our model are indeed more discriminative. This also well validates the effectiveness of regularizing the hidden factors by the orthogonality-based loss function. Then, we explored the effect of the number of factors on the model performance in both compatibility prediction task and fill-in-the-blank task. As can be seen from Figure 3, we noticed that the model performance does not monotonically increase with increasing number of factors, but first increases until K grows up to 5, and then decreases with K further grows. This demonstrates that our OCM-CF is able to achieve the optimal performance with only a few hidden factor subspaces. Nevertheless, introducing too many hidden factors, like $K = 20$, may incorporate noise, resulting in the performance degradation.



(a) with complementarity loss. **(b) without complementarity loss.**

Figure 2: Visualization of the latent outfit representations obtained by our model.



(a) K on Compat. **(b) K on FITB**

Figure 3: Effect of the number of hidden factors, i.e., K, on both compatibility prediction and FITB tasks.

4.5 On Case Study (RQ3)

To gain a more intuitive understanding of our model, we conducted the case study on two tasks: similar outfit retrieval and complementary fashion item retrieval.

4.5.1 Similar Outfit Retrieval. To illustrate the effectiveness of the outfit representation learned by our model, we investigated the performance of our model in the task of similar outfit retrieval, which aims to retrieve similar outfits for a given query outfit. We argued that similar outfits tend to share the common prominent features. Instead of using all factor-oriented outfit representations, we particularly adopted the outfit representation corresponding to the highest compatibility score, i.e., $\mathbf{p}^{k^*}$, where $k^* = \arg\max_{k} \{\mathbf{W}_s^k \mathbf{p}^k|_{k=1}^{K}\}$, and employed the cosine similarity between the query outfit and each candidate outfit to retrieve the similar outfit for the query outfit. Towards the comprehensive evaluation, we studied the similar outfit retrieval task in two scenarios: 1) the candidate outfits have

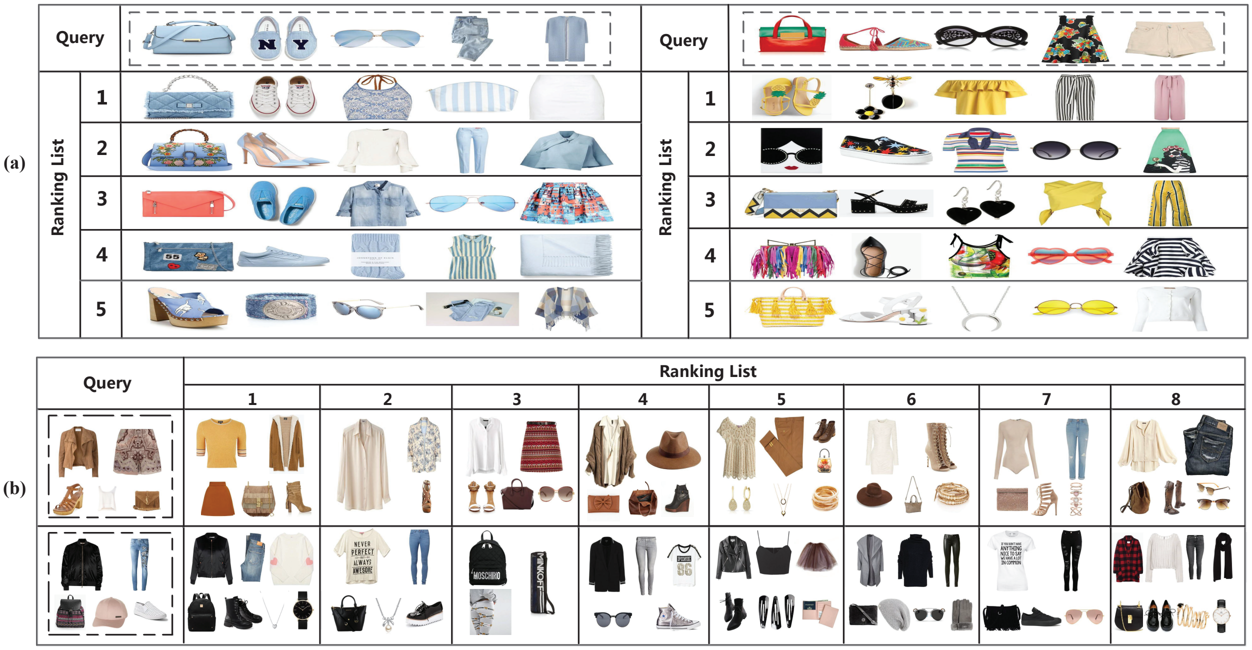


Figure 4: Illustration of the similar outfit retrieval results with testing samples in two scenarios.

the same length, i.e., the same number of composition items with the query outfit, and 2) the candidate outfits have random length. In this part, we directly employed the test dataset of the compatibility prediction as the set of candidate outfits. Figure 4 illustrates the retrieval results of two testing outfits in each scenario. For the first scenario, from the left example in Figure 4a, we observed that the retrieved outfits share the blue main color, and the fresh style, while from the right example, we noticed that the returned outfits also possess the high similarity with the query outfit, such as the summer style, a variety of color and pattern, and the item categories. Similar observations can be also obtained from Figure 4b, where the length of the retrieved outfits is not restricted. In general, these observations reflect the effectiveness of the factor-oriented outfit representation learned by our model, and the benefit of exploring the hidden complementary factors to capture the discriminative feature of the outfit.

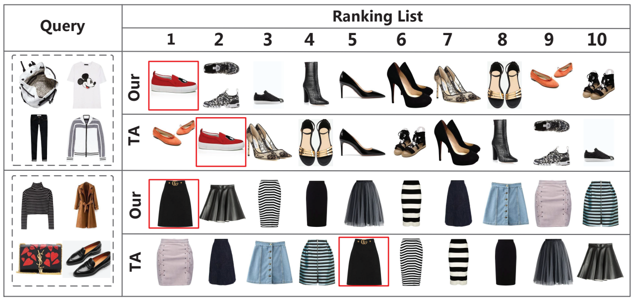


Figure 5: Illustration of the complementary item retrieval results. Positive items are highlighted in red boxes.

4.5.2 Complementary Fashion Item Retrieval. Similar to existing studies [3, 30], we also presented the qualitative results of our model in the complementary item retrieval task, where the candidate set comprises a target item as well as nine negative items. Moreover, we adopted two negative item sampling protocols: the negative items are randomly selected from items with the same coarse-grained category with the target item, and 2) the negative items are randomly selected from items with the same fine-grained category with the target item, which corresponds to a more challenging task. In addition, we adopted the best baseline on the complementary item retrieval task, i.e., Type-aware, for comparison. Due to the limited space, we only exhibited one example for each scenario in Figure 5. As can be seen, our OCM-CF is able to rank the target items at the top places, outperforming the Type-aware.

5 CONCLUSION AND FUTURE WORK

In this work, we present a novel outfit compatibility modeling scheme via complementary factorization, named as OCM-CF, which seamlessly unifies the context-aware outfit representation learning and hidden complementary factors learning in the context of outfit compatibility modeling. Extensive experiments have been conducted on two real-world datasets, and the encouraging experiment results validate the superiority of our proposed model and the importance of each component. In addition, we notice that the global outfit representation indeed models one compatible factor by considering all items of the outfit with a comprehensive perspective and the proposed orthogonality-based complementarity regularization is able to make the factor-oriented outfit representation discriminative. One limitation of our work is that currently we only focus on the outfit compatibility from a general standard, overlooking the user preferences. In the future, we plan to explore the personalized outfit compatibility modeling.

ACKNOWLEDGMENTS

This work is supported by the Shandong Provincial Natural Science Foundation, No.:ZR2019JQ23; Kuaishou, and CCF-Baidu Open Fund, No.:CCF-BAIDU OF2020019.

REFERENCES

[1] Guillem Cucurull, Perouz Taslakian, and David Vázquez. 2019. Context-Aware Visual Compatibility Prediction. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. IEEE, 12617–12626.

[2] Zeyu Cui, Zekun Li, Shu Wu, Xiaoyu Zhang, and Liang Wang. 2019. Dressing as a Whole: Outfit Compatibility Learning Based on Node-wise Graph Neural Networks. In *Proceedings of the International World Wide Web Conference*. ACM, 307–317.

[3] Xue Dong, Jianlong Wu, Xuemeng Song, Hongjun Dai, and Liqiang Nie. 2020. Fashion Compatibility Modeling through a Multi-modal Try-on-guided Scheme. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 771–780.

[4] M. Gori, G. Monfardini, and F. Scarselli. 2005. A new model for learning in graph domains. *Proceedings of the IEEE International Joint Conference on Neural Networks* 2 (2005), 729–734.

[5] William L. Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive Representation Learning on Large Graphs. In *Proceedings of the Advances in Neural Information Processing Systems*. NIPS, 1024–1034.

[6] Xintong Han, Zuxuan Wu, Yu-Gang Jiang, and Larry S. Davis. 2017. Learning Fashion Compatibility with Bidirectional LSTMs. In *Proceedings of the International ACM Conference on Multimedia*. ACM, 1078–1086.

[7] Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. IEEE, 770–778.

[8] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yong-Dong Zhang, and Meng Wang. 2020. LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 639–648.

[9] Yupeng Hu, Meng Liu, Xiaobin Su, Zan Gao, and Liqiang Nie. 2021. Video Moment Localization via Deep Cross-Modal Hashing. *IEEE Transactions on Image Processing* 30 (2021), 4667–4677.

[10] Yupeng Hu, Liqiang Nie, Meng Liu, Kun Wang, Yinglong Wang, and Xiansheng Hua. 2021. Coarse-to-Fine Semantic Alignment for Cross-Modal Moment Localization. *IEEE Transactions on Image Processing* 30 (2021), 5933–5943.

[11] Wang-Cheng Kang, Chen Fang, Zhaowen Wang, and Julian J. McAuley. 2017. Visually-Aware Fashion Recommendation and Design with Generative Image Models. In *Proceedings of the IEEE International Conference on Data Mining*. IEEE, 207–216.

[12] Thomas Kipf and M. Welling. 2016. Variational Graph Auto-Encoders. In *NIPS Workshop on Bayesian Deep Learning*.

[13] Thomas N. Kipf and Max Welling. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *Proceedings of the International Conference on Learning Representations*. OpenReview.net.

[14] Xingchen Li, Xiang Wang, Xiangnan He, Long Chen, Jun Xiao, and Tat-Seng Chua. 2020. Hierarchical Fashion Graph Network for Personalized Outfit Recommendation. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 159–168.

[15] Yujia Li, Daniel Tarlow, Marc Brockschmidt, and Richard S. Zemel. 2016. Gated Graph Sequence Neural Networks. In *Proceedings of the International Conference on Learning Representations*.

[16] Zhi Li, Bo Wu, Qi Liu, Likang Wu, Hongke Zhao, and Tao Mei. 2020. Learning the Compositional Visual Coherence for Complementary Recommendations. In *Proceedings of the International Joint Conference on Artificial Intelligence*. ijcai.org, 3536–3543.

[17] Lizi Liao, Yunshan Ma, Xiangnan He, Richang Hong, and Tat-Seng Chua. 2018. Knowledge-aware Multimodal Dialogue Systems. In *Proceedings of the International ACM Conference on Multimedia*. ACM, 801–809.

[18] Jinhuan Liu, Xuemeng Song, Zhaochun Ren, Liqiang Nie, Zhaopeng Tu, and Jun Ma. 2020. Auxiliary Template-Enhanced Generative Compatibility Modeling. In *Proceedings of the International Joint Conference on Artificial Intelligence*. ijcai.org, 3508–3514.

[19] M. Liu, L. Nie, X. Wang, Q. Tian, and B. Chen. 2019. Online Data Organizer: Micro-Video Categorization by Structure-Guided Multimodal Dictionary Learning. *IEEE Transactions on Image Processing* 28, 3 (2019), 1235–1247.

[20] Zhi Lu, Yang Hu, Yunchao Jiang, Yan Chen, and Bing Zeng. 2019. Learning Binary Code for Personalized Fashion Recommendation. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. IEEE, 10562–10570.

[21] Andrew L Maas, Awni Y Hannun, and Andrew Y Ng. 2013. Rectifier nonlinearities improve neural network acoustic models. In *Proceedings of the International Conference on Machine Learning*. 3.

[22] Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of machine learning research* 9 (2008), 2579–2605.

[23] Takuma Nakamura and Ryosuke Goto. 2018. Outfit Generation and Style Extraction via Bidirectional LSTM and Autoencoder. *CoRR* abs/1807.03133 (2018).

[24] Liqiang Nie, Wenjie Wang, Richang Hong, Meng Wang, and Qi Tian. 2019. Multimodal Dialog System: Generating Responses via Adaptive Decoders. In *Proceedings of the International ACM Conference on Multimedia*. ACM, 1098–1106.

[25] Gabriele Prato, Federico Sallemi, Paolo Cremonesi, Mario Scriminaci, Stefan Gudmundsson, and Silvio Palumbo. 2020. Outfit Completion and Clothes Recommendation. In *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*. ACM, 1–7.

[26] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian Personalized Ranking from Implicit Feedback. In *Proceedings of the International Conference on Uncertainty in Artificial Intelligence*. AUAI Press, 452–461.

[27] Yu Rong, Wenbing Huang, Tingyang Xu, and Junzhou Huang. 2020. DropEdge: Towards Deep Graph Convolutional Networks on Node Classification. In *Proceedings of the International Conference on Learning Representations*. OpenReview.net.

[28] Olga Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Zhiheng Huang, A. Karpathy, A. Khosla, Michael S. Bernstein, A. Berg, and Li Fei-Fei. 2015. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision* 115 (2015), 211–252.

[29] Xuemeng Song, Fuli Feng, Jinhuan Liu, Zekun Li, Liqiang Nie, and Jun Ma. 2017. NeuroStylist: Neural Compatibility Modeling for Clothing Matching. In *Proceedings of the International ACM Conference on Multimedia*. ACM, 753–761.

[30] Xuemeng Song, Xianjing Han, Yunkai Li, Jingyuan Chen, Xin-Shun Xu, and Liqiang Nie. 2019. GP-BPR: Personalized Compatibility Modeling for Clothing Matching. In *Proceedings of the International ACM Conference on Multimedia*. ACM, 320–328.

[31] Reuben Tan, Mariya I. Vasileva, Kate Saenko, and Bryan A. Plummer. 2019. Learning Similarity Conditions Without Explicit Supervision. In *IEEE/CVF International Conference on Computer Vision*. IEEE, 10372–10381.

[32] Mariya I. Vasileva, Bryan A. Plummer, Krishna Dusad, Shreya Rajpal, Ranjitha Kumar, and David A. Forsyth. 2018. Learning Type-Aware Embeddings for Fashion Compatibility. In *Proceedings of the European Conference on Computer Vision*. Springer, 405–421.

[33] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Proceedings of the Advances in Neural Information Processing Systems*. NIPS, 5998–6008.

[34] Andreas Veit, Serge J. Belongie, and Theofanis Karaletsos. 2017. Conditional Similarity Networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. IEEE, 1781–1789.

[35] Andreas Veit, Balazs Kovacs, Sean Bell, Julian J. McAuley, Kavita Bala, and Serge J. Belongie. 2015. Learning Visual Clothing Style with Heterogeneous Dyadic Co-Occurrences. In *Proceedings of the IEEE International Conference on Computer Vision*. IEEE, 4642–4650.

[36] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph Attention Networks. In *Proceedings of the International Conference on Learning Representations*. OpenReview.net.

[37] Xiang Wang, Hongye Jin, An Zhang, Xiangnan He, Tong Xu, and Tat-Seng Chua. 2020. Disentangled Graph Collaborative Filtering. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 1001–1010.

[38] Xin Wang, Bo Wu, Yueqi Zhong, Wei Hu, and Jan Zahálka. 2020. Reproducibility Companion Paper: Outfit Compatibility Prediction and Diagnosis with Multi-Layered Comparison Network. In *Proceedings of the International ACM Conference on Multimedia*. ACM, 4439–4443.

[39] Yinwei Wei, Xiang Wang, Liqiang Nie, Xiangnan He, Richang Hong, and Tat-Seng Chua. 2019. MMGCN: Multi-modal graph convolution network for personalized recommendation of micro-video. In *Proceedings of the International ACM Conference on Multimedia*. ACM, 1437–1445.

[40] Xun Yang, Xiaoyu Du, and Meng Wang. 2020. Learning to Match on Graph for Fashion Compatibility Modeling. In *Proceedings of the AAAI Conference on Artificial Intelligence*. AAAI Press, 287–294.

[41] Xuewen Yang, Dongliang Xie, Xin Wang, Jiangbo Yuan, Wanying Ding, and Pengyun Yan. 2020. Learning Tuple Compatibility for Conditional Outfit Recommendation. In *Proceedings of the International ACM Conference on Multimedia*. ACM, 2636–2644.

[42] Cong Yu, Yang Hu, Yan Chen, and Bing Zeng. 2019. Personalized Fashion Design. In *Proceedings of the IEEE International Conference on Computer Vision*. IEEE, 9045–9054.

[43] Hanwang Zhang, Zheng-Jun Zha, Yang Yang, Shuicheng Yan, Yue Gao, and Tat-Seng Chua. 2013. Attribute-augmented semantic hierarchy: towards bridging semantic gap and intention gap in image retrieval. In *Proceedings of the International ACM Conference on Multimedia*. ACM, 33–42.

[44] Zhaolong Zhang, Yuejie Zhang, Rui Feng, Tao Zhang, and Weiguo Fan. 2020. Zero-Shot Sketch-Based Image Retrieval via Graph Convolution Network. In *Proceedings of the International Joint Conference on Artificial Intelligence*. AAAI Press, 12943–12950.

[45] Xiang Zhou, Fumin Shen, Li Liu, Wei Liu, Liqiang Nie, Yang Yang, and Heng Tao Shen. 2020. Graph Convolutional Network Hashing. *IEEE Transactions on Cybernetics* 50, 4 (2020), 1460–1472.