Tops, Bottoms, and Shoes: Building Capsule Wardrobes via Cross-Attention Tensor Network

Huiyuan Chen
hchen@visa.com
Visa Research, Palo Alto
USA

Yusan Lin
yusalin@visa.com
Visa Research, Palo Alto
USA

Fei Wang
feiwang@visa.com
Visa Research, Palo Alto
USA

Hao Yang
haoyang@visa.com
Visa Research, Palo Alto
USA

ABSTRACT

Fashion is more than Paris runways. Fashion is about how people express their interests, identity, mood, and cultural influences. Given an inventory of candidate garments from different categories, how to assemble them together would most improve their fashionability? This question presents an intriguing visual recommendation challenge to automatically create capsule wardrobes. Capsule wardrobe generation is a complex combinatorial problem that requires the understanding of how multiple visual items interact. The generative process often needs fashion experts to manually tease the combinations out, making it hard to scale.

We introduce TensorNet, an approach that captures the key ingredients of visual compatibility among tops, bottoms, and shoes. TensorNet aims to provide actionable advice for *full-body* clothing outfits that mix and match well. Our TensorNet consists of two core modules: a *Cross-Attention Message Passing* module and a *Wide&Deep Tensor Interaction* module. As such, TensorNet is able to characterize the local region-based patterns as well as the global compatibility of the entire outfits. Our experimental results on the real-word datasets indicate that the proposed method is capable of learning visual compatibility and outperforms all the baselines. TensorNet opens up opportunities for fashion designers to narrow down the search space for multi-clothes combinations.

KEYWORDS

Fashion Recommendation, Neural Tensor Network, Cross-Attention, Linear Attention

ACM Reference Format:
Huiyuan Chen, Yusan Lin, Fei Wang, and Hao Yang. 2021. Tops, Bottoms, and Shoes: Building Capsule Wardrobes via Cross-Attention Tensor Network . In *Fifteenth ACM Conference on Recommender Systems (RecSys '21), September 27-October 1, 2021, Amsterdam, Netherlands.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3460231.3474258

RecSys '21, September 27-October 1, 2021, Amsterdam, Netherlands

ACM ISBN 978-1-4503-8458-2/21/09…$15.00
https://doi.org/10.1145/3460231.3474258

1 INTRODUCTION

"*Fashion has always been a repetition of ideas, but what makes it new is the way you put it together.*" — Carolina Herrera

With coherent outfit combinations, not only can one accurately convey their social classes, but also express their subcultural identities [17]. The underlying rules of how to put clothing items together date all the way back to the 18-th century. Concrete suggestions for women, such as "the hoods may be made of satin to match the lining and frill of the jacket, but should be lined with fine white cashmere", were given as specific dressing instructions [3].

With consumers' increasing desire of wearing stylish outfits, the online fashion markets are expected to reach $872 billion USD in 2023, from $533 billion USD in 2018[1]. However, searching for the right outfit compositions is not trivial. Survey shows that consumers experience on average 36 times "wardrobe panic" annually, when they struggle to pair items together from their wardrobes to compose a nice outfit[2]. This question presents an intriguing visual recommendation challenge to automatically create capsule wardrobes. For instance, given an inventory of candidate garments from different categories, how should one assemble them together to achieve the most fashionability?

To answer above question, visually-aware recommender systems have been designed to provide a set of clothing items such that these items are both visually compatible and functionally irredundant [8, 11, 18, 24, 30, 39]. These methods map items (e.g., using CNNs) into an embedding space, where similar items are embedded nearby, and items that are different are widely separated. The compatibility of visual embeddings are then determined by learning various loss functions. For example, Han et al. [11] presented a bidirectional LSTM model to predict the next item according item compatibility relationships on a visual embedding space. Chen et al. [8] obtained a visual embedding space by minimizing a triplet loss, such that two items from the same category are close to each other. These strategies have great success in capturing global compatibility among items. However, the discriminative local regions of clothes are often ignored [22]. Unlike generic objects, the fashion styles of clothes are typically represented by some attributes in local regions, such as collar, pocket, sleeve, V-neck, etc. These

[1]https://www.salecycle.com/blog/featured/online-fashion-retail-11-essential-statistics/
[2]https://www.trunkclub.com/press/news/new-trunk-club-survey-finds-americans-experience-wardrobe-panic-36-times-annually

detailed designs within items also affect the overall outfit quality significantly [1]. Naturally, human vision verifies whether two clothing items are compatible in terms of both their global features (e.g., color, shape) as well as the local features (e.g., sleeve, logos). Therefore, we believe that learning global and local compatibility jointly can bring out more insights to the outfit quality.

Present Work. Generally speaking, the foundation of outfits contains tops, bottoms and shoes. Missing any of these pieces would make the outfit incomplete, while it is acceptable when an outfit doesn't include an outerwear, accessory, etc. With this important concept in mind, we cast capsule wardrobe generation as a combinatorial problem of selecting a candidate triplet (top, bottom, shoe) to form an outfit that maximizes its compatibility. In this paper, we proposed a neural Tensor Network (TensorNet) that captures key ingredients of visual compatibility among tops, bottoms, and shoes, as shown in Fig. 1. Tensors [21], as high-dimensional generalizations of matrices, have shown great promise in context-aware recommendations [19, 39]. Our capsule wardrobe generation problem is well-suited for tensor (e.g., a third-order $top \times bottom \times shoe$ tensor) that captures multi-way interactions among tops, bottoms, and shoes. Fig. 2 illustrates the overall architecture of the proposed TensorNet. Our model mainly addresses four key challenges in fashion recommendations:

- **Local Compatibility:** Local regions express the fashion style of clothes in details. We propose a region-wise feature map approach that extracts regional features from the given images of clothing items. The extracted feature maps achieve precise matching of clothing components in fashion design. For example, ruffle designs on tops usually are compatible with more feminine design of shoes, such as heels and slingback. While they are not compatible with more masculine designs of shoes, such as combat boots.
- **Cross-Attention Message Passing:** Designing a good outfit composition requires judging how well-coordinated a top-bottom-shoe triplet is. Intuitively, not all regions contribute equally to the overall aesthetics and compatibility. Therefore, we design a *cross-attention message passing* mechanism to exploit how the visual messages pass within the path: $top \leftrightarrow bottom \leftrightarrow shoe$, enabling region-wise information exchange between nearby clothes, such as $top \leftrightarrow bottom$ and $bottom \leftrightarrow shoe$. Nevertheless, the standard attention mechanism requires quadratic complexity in order to compute affinities of all the region pairs. To address this bottleneck, we put forward a simple but effective method to linearize attentions, which reduces the complexity.
- **Visual Gated Units:** In general, the messages (e.g., visual signals from bottom to top) can contain irrelevant or negative (mismatched) signals, which need to be filtered out during pairwise matching. For example, the studs on a pair of boots should pass more visual signals to the distressed on a pair of jeans, but less signals to the collar on a shirt. To better align knowledge between two nearby clothing items, we further adopt learnable gated units to allow the model to adaptively control what information should be propagated across visual paths.
- **Global Compatibility:** After obtaining more fine-grained embeddings of tops, bottoms, and shoes, we introduce a neural tensor layer to measure their global compatibility. It can learn higher-order and non-linear feature interactions for real-world data by using wide&deep learning strategies. To learn the parameters of TensorNet, a K-pair contrastive loss is also applied to identify a positive sample from multiple negative samples, allowing to reach better local optima.

We evaluate our proposed model on two large fashion visual datasets: Polyvore and iFashion. By comparing with numerous state-of-the-arts, we show that our proposed model outperforms them significantly. TensorNet aims to provide actionable advice for *full-body* clothing outfits that mix and match well.

Lastly, it is worth mentioning that our TensorNet can be easily extended to an arbitrary-order tensor network for studying compatibility of a capsule wardrobe that contains clothes from rich categories. For example, when accessories (e.g., hat, bag) are available, one can construct a fifth order $hat \times top \times bottom \times shoe \times bag$ tensor to puzzle together an outfit. More importantly, our TensorNet is able to provide personalized fashion outfits when associating with user behavior data, i.e., exploiting a fourth-order $user \times top \times bottom \times shoe$ tensor. We leave these extensions in the future. Taken together, this study shows the ability to build capsule wardrobes via tensor network, and opens up new opportunities for fashion designers to narrow down the search space for multi-clothes combinations.

2 RELATED WORK

2.1 Fashion Compatibility Recommendation

In the past few years, there has been a surge of models proposed to learn the compatibility of fashion outfits [8, 11, 13, 18, 24, 30, 39]. These methods aim to exploit the two key properties: *compatibility* and *personalization*. Compatibility indicates items from different types can go together in an outfit, while personalization shows how candidate outfits meet consumers' personal tastes. For example, Xu et al. [8] proposed a personalized outfit model, which connected user preferences and outfits within Transformer network. Yin et al. [38] proposed to use Bayesian Personalized Ranking to learn the compatibility given pairs of fashion items by leveraging color histogram information. Lin et al. [24] introduced OutfitNet in two stages: first studying the item compatibility and then recommending outfits to users to meet their personal tastes. Nevertheless, many of them take a complete outfit image as an input and do not disentangle the region-wise features included in the outfit. Moreover, private user data, like purchase history, are always required for personalization. However, personal data collection is becoming difficult due to several laws protecting user privacy, such as the General Data Protection Regulation (GDPR)[3].

Here we focus on the compatibility problem, and aim to learn high-order and nonlinear feature interactions of a set of garments that can be assembled into many compatible outfits. We are aware that certain techniques (e.g., federated learning) can be used for personalization without compromising user privacy. We leave this extension in the future.

[3]https://recsys.acm.org/recsys19/keynotes/

2.2 Tensor Factorization

Tensors are powerful tools to model multi-modal information [21]. Tensor factorizations, built upon the multi-linear tensor algebra, seek to fill the unobserved entries of partially observed tensors, which have been widely used in recommender systems [6, 19, 28, 39]. For example, Rendle et al. [28] introduced a tensor method to exploit the pairwise interactions between users, items and tags. Yu et al. [39] incorporated aesthetic features into a tensor factorization model to capture the aesthetic preference of consumers. Inspired by these tensor-based models, we aim to build a new tensor model to study the triple-wise interactions among tops, bottoms, and shoes for outfit recommendation.

In addition to multi-linear tensor models, non-linear tensor factorizations have gained attention due to their effectiveness at learning complex patterns [25, 34, 36]. Xu et al. [36] introduced a series of Gaussian kernels to capture non-linear relationships. Motivated by the deep neural networks, Liu et al. [25] proposed to use a convolutional neural network to mine sparse tensor data, while Wu et al. [34] replaced the multi-linear operations with multi-layer perceptrons to model complex relationships. In this work, our proposed TensorNet, being more general, further increased the expressive power of non-linear tensor models by using the wide&deep learning strategies, which have been successfully applied to two-dimensional matrix completion in recommender systems [9, 14].

2.3 Cross-Attention Network

Attention mechanisms aim to highlight important local regions to extract more meaningful features, which have been successfully used in various visual and textual tasks, including machine translation [31], image classification [16], and fashion recommendation [8]. Recently, attention mechanisms have also been applied to investigate the knowledge alignment between two objects in cross-modal tasks [16, 23, 32, 33, 37]. The key idea of cross-attention mechanisms is to enhance the compatibility between attention selections and feature representations, given their semantic dependencies. For instance, Lee et al. [23] presented a stacked cross-attention network to infer the latent semantic alignment between visual regions and words in sentences, making image-text matching more interpretable. Hou et al. [16] introduced a cross-attention module to learn the semantic relevance between unseen classes and query features in few-shot classifications. In this study, we generate cross-attention maps for exploiting how the visual messages pass within the path: *top* $\leftrightarrow$ *bottom* $\leftrightarrow$ *shoe*, allowing the model to capture the fine-grained interplay between neighboring clothes, in particular *top* $\leftrightarrow$ *bottom* and *bottom* $\leftrightarrow$ *shoe*.

One major challenge of cross-attention mechanisms is its quadratic time and memory complexity for computing the softmax attentions, which precludes their usage in settings with limited computational resources. Inspired by recent kernelizable attention techniques [10, 20, 26, 40], we further linearize the regular softmax attentions to reduce both time and space complexity of cross-attention modules, from quadratic to linear.

3 THE PROPOSED MODEL

In this section, we first describe the region-wise features for each component of an outfit. Then, we propose a Tensor Network (TensorNet) to measure the compatibility of any triplet (top, bottom,

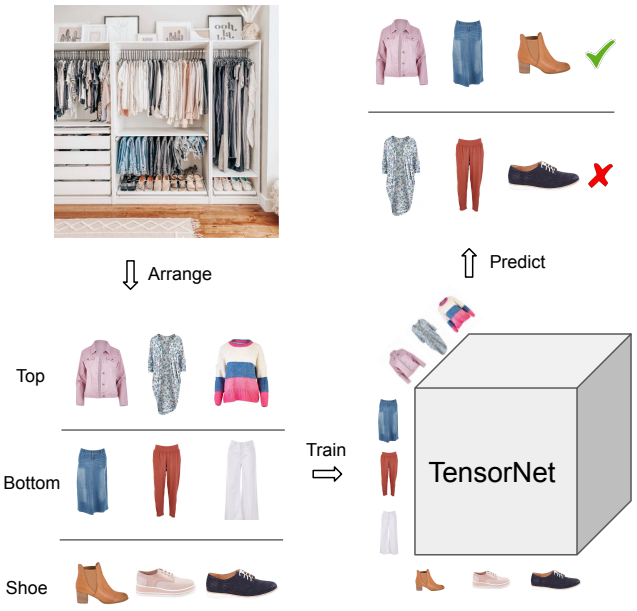

Figure 1: Given a capsule wardrobe containing three categories: tops, bottoms, and shoes, our approach aims to recommend a compatible outfit.

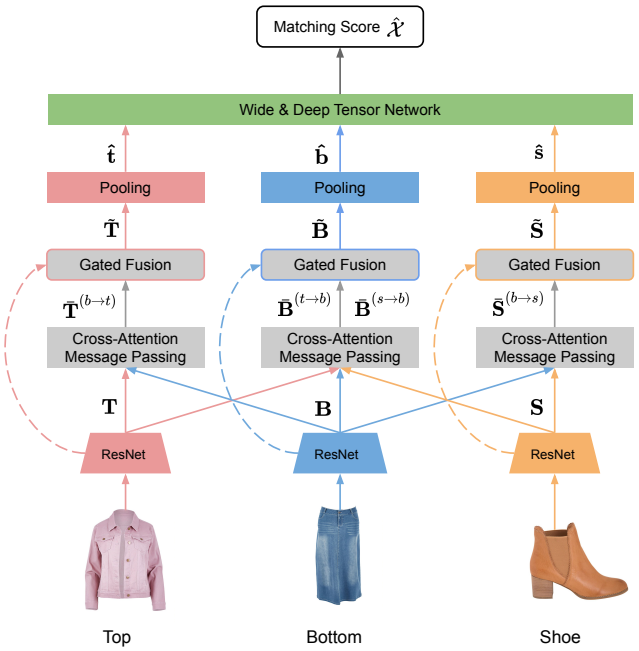

Figure 2: The overall architecture of our TensorNet to measure the compatibility and fashionability for a triplet (top, bottom, shoe) given an outfit.

shoe) in an outfit (Fig. 2). Our TensorNet consists of two core modules: a *Cross-Attention Message Passing* module and a *Wide&Deep Tensor Interaction* module. As such, TensorNet can characterize both the local region-based patterns and the global compatibility of the entire outfits. Additionally, we put forward a K-pair loss function to discriminate a positive sample from multiple negative samples, which improves its generalization performance.

3.1 Task Description

A capsule wardrobe contains a subset of garments that mix and match well. In this paper, we cast capsule creation as the problem of selecting triplets among tops, bottoms, and shoes that maximize their compatibility and fashionability. To be specific, we use a $top \times bottom \times shoe$ tensor $\mathcal{X} \in \mathbb{R}^{M \times N \times L}$ to indicate the outfit events, where M, N, and L denote the number of tops, bottoms, and shoes, respectively. An entry $\mathcal{X}_{pqr} = 1$ if the triplet (e.g., a top image $I_t^{(p)}$, a bottom image $I_b^{(q)}$, and a shoe image $I_s^{(r)}$) creates a compatible outfit, $\mathcal{X}_{pqr} = 0$ otherwise. Our goal is to predict an outfit compatibility score for those triplets that have not yet been observed (i.e., zero elements), which can be used for outfit recommendations.

3.2 Region-Wise Feature Map

Beyond global colors and shapes, the visual style is another key ingredient to describe clothing. Nevertheless, the visual style is generally determined by some local attributes, such as collars, pockets, sleeves, etc [1]. Inspired by recent advances in image reasoning [2, 27], we build up a visual reasoning model to enhance the region-based representations by considering the semantic relationships among the clothing regions.

We adopt ResNet50 [12] (pretrained on ImageNet) to extract visual features from tops, bottoms, and shoes. Given a top image I_t[4], we obtain its region-wise feature maps $\mathbf{T}' = [\mathbf{t}'_1, \ldots, \mathbf{t}'_{R_t}] \in \mathbb{R}^{D \times R_t}$ from intermediate convolutional layers (e.g., conv5_3), where R_t is the number of regions of the top and $\mathbf{t}'_r \in \mathbb{R}^D$ is the feature vector corresponding to the r-th region. These ResNet feature maps haven been shown to be able to capture key context from local regions, with strong performance and transferability [27]. Similarly, the feature maps for a bottom image I_b and a shoe image I_s are denoted as $\mathbf{B}' = [\mathbf{b}'_1, \ldots, \mathbf{b}'_{R_b}] \in \mathbb{R}^{D \times R_b}$ and $\mathbf{S}' = [\mathbf{s}'_1, \ldots, \mathbf{s}'_{R_s}] \in \mathbb{R}^{D \times R_s}$, respectively. Due to the limited size of our datasets, we freeze the weights of ResNet50 and apply three fully-connected networks $g(\Theta; \cdot)$ to transform the features into d-dimensional embeddings:

$$\mathbf{T} = g(\Theta_t; \mathbf{T}'), \quad \mathbf{B} = g(\Theta_b; \mathbf{B}'), \quad \mathbf{S} = g(\Theta_s; \mathbf{S}'), \tag{1}$$

where $\mathbf{T} \in \mathbb{R}^{d \times R_t}$, $\mathbf{B} \in \mathbb{R}^{d \times R_b}$, and $\mathbf{S} \in \mathbb{R}^{d \times R_s}$ are new feature maps for the top, bottom, and shoe, respectively. l_2 normalization is also applied on their columns to improve training stability. Taking advantage of these feature maps, we next introduce a Cross-Attention Message Passing strategy to locally detect region-wise feature interactions among tops, bottoms, and shoes.

[4]To ease the explanation, we simply discard the subscript: $I_t^{(p)} \to I_t$, $I_b^{(q)} \to I_b$, and $I_s^{(r)} \to I_s$ for top p, bottom q, and shoe r.

3.3 Cross-Attention Message Passing

Given an outfit, one natural way to measure the compatibility is to exploit how the messages pass within the path: $top \leftrightarrow bottom \leftrightarrow shoe$, where bottoms serve as bridges connecting tops and shoes. As such, we aim to extract the most salient feature matching from two sub-paths: $top \leftrightarrow bottom$ and $bottom \leftrightarrow shoe$. And we leave the flexibility to combine tops and shoes in versatile ways to create as many stylish outfits as possible.

The cross-attention module contains two parts: 1) Attentive pairwise matching, which calculates how the aggregated messages can be propagated from one object to another. A linearized softmax attention is also proposed to reduce time complexity. 2) Visual fusion gate units, which are capable of regulating how much of the messages propagates between two objects. In what follows, we will describe the details of these two parts.

3.3.1 ***Attentive Pairwise Matching.*** Inspired by the recent cross-attention mechanism [16, 23, 32, 33], we generate cross attention maps for each sub-path to highlight the regions of interest, making extracted features more coordinated to each other. In particular, we define four pairwise message passing routines: $bottom \to top$, $top \to bottom$, $bottom \to shoe$, and $shoe \to bottom$. These routines enable the information flow across full-body outfits and select the most salient feature maps to show their local compatibility. Note that, the routines $bottom \to top$ and $top \to bottom$ are asymmetric due to the cross-attention mechanism [23]. We next propose a simple but efficient attention method to speed up the message passing in details.

1. Message Passing for Bottom→Top: Given the region-wise features $\mathbf{T} = [\mathbf{t}_1, \ldots, \mathbf{t}_{R_t}] \in \mathbb{R}^{d \times R_t}$ (top) and $\mathbf{B} = [\mathbf{b}_1, \ldots, \mathbf{b}_{R_b}] \in \mathbb{R}^{d \times R_b}$ (bottom) from Eq. (1), we aim to enrich $\mathbf{T}$ and $\mathbf{B}$ by transferring region-to-region messages between top and bottom. Formally, we first compute the bottom-top affinity matrix $\mathbf{A} \in \mathbb{R}^{R_b \times R_t}$ for all possible pairs of regions [23, 31]:

$$\mathbf{A}_{ij} = \mathbf{b}_i^T \cdot \mathbf{t}_j, \quad 1 \le i \le R_b, 1 \le j \le R_t, \tag{2}$$

where $\mathbf{A}$ is the cosine similarity[5] that represents the affinity between the i-th region of bottom and the j-th region of top. To derive the cross-attention for bottom-to-top, we adopt a weighted combination:

$$\bar{\mathbf{t}}_j^{(b \to t)} = \sum_{i=1}^{R_b} \alpha_{ij}^{(b \to t)} \mathbf{b}_i, \quad \text{and } \alpha_{ij}^{(b \to t)} = \frac{e^{\mathbf{A}_{ij}/\tau}}{\sum_{i=1}^{R_b} e^{\mathbf{A}_{ij}/\tau}}, \tag{3}$$

where $\alpha^{(b \to t)}$ is the bottom-to-top attention matrix by softmax normalizing the affinity matrix $\mathbf{A}$ across the *bottom*-dimension; τ is the temperature determining how flat the softmax is; $\bar{\mathbf{t}}_j^{(b \to t)}$ denotes the bottom features attended by the j-th region of top, and $\bar{\mathbf{T}}^{(b \to t)} = [\bar{\mathbf{t}}_1^{(b \to t)}, \ldots, \bar{\mathbf{t}}_{R_t}^{(b \to t)}] \in \mathbb{R}^{d \times R_t}$ can be regarded as the aggregated messages to be passed from bottom to top.

2. Linearized Attention: The Eq. (3) is a standard definition of attention, where the computational cost of softmax attention is quadratic for all queries ($O(dR^2)$ in our case, where $R = \min\{R_t, R_b\}$). The same is true for the memory requirements since the *full* attention matrix must be stored explicitly to compute the gradients in

[5]Here we have $\|\mathbf{t}_i\|_2 = \|\mathbf{b}_j\|_2 = 1$ due to l_2 normalization in Eq. (1).

the backpropagation. Combing Eq. (2) and Eq. (3), we can write a generalized attention form as [20]:

$$\bar{\mathbf{t}}_j^{(b\to t)} = \frac{\sum_{i=1}^{R_b} \mathtt{sim}(\mathbf{b}_i, \mathbf{t}_j)\mathbf{b}_i}{\sum_{i=1}^{R_b} \mathtt{sim}(\mathbf{b}_i, \mathbf{t}_j)}, \tag{4}$$

where $\mathtt{sim}(\mathbf{b}_i, \mathbf{t}_j)$ can be any similarity function with non-negative property. Eq. (4) is equivalent to Eq. (3) if we substitute the similarity function with $\mathtt{sim}(\mathbf{b}_i, \mathbf{t}_j) = e^{\mathbf{b}_i^T \cdot \mathbf{t}_j/\tau}$.

Several recent studies [10, 20, 26, 40] have attempted to reduce the complexity of attention mechanisms, from quadratic to linear, by using kernel functions. One elegant approach is the *Linear Transformer* [20], which designs a kernel function as:

$$\mathtt{sim}(\mathbf{b}_i, \mathbf{t}_j) = \phi(\mathbf{b}_i)^T \phi(\mathbf{t}_j), \quad \text{and } \phi(\mathbf{x}) = \mathrm{elu}(\mathbf{x}) + 1, \tag{5}$$

where $\mathrm{elu}(\cdot)$ is the exponential linear unit. The design choice of $\mathrm{elu}(\cdot)$ is motivated by non-zero gradients on the negative parts. The complexity reduction is mainly due to a linearization of the softmax. However, the approximation error can be large in some cases [4, 35]. Softmax linearization techniques for Transformers are still under-explored. The existing approximation are either oversimplified [20] or mathematically well explained but very complex [10, 26].

Alternatively, we propose a simple but effective approximated function by using the Talyor Series, i.e., $e^x = \sum_{n=0}^{\infty} \frac{1}{n!} x^n$. Given the $\mathtt{sim}(\mathbf{b}_i, \mathbf{t}_j) = e^{\mathbf{b}_i^T \cdot \mathbf{t}_j/\tau}$ in the softmax attention, its Talyor Series are:

$$e^{\mathbf{b}_i^T \cdot \mathbf{t}_j/\tau} = 1 + \mathbf{b}_i^T \cdot \mathbf{t}_j/\tau + \frac{(\mathbf{b}_i^T \cdot \mathbf{t}_j/\tau)^2}{2!} + \cdots,$$

Intuitively, the mapping function $\phi(\cdot)$ that corresponds to the exponential function in the softmax should be infinite dimensional, which makes the linearization of *exact* softmax attention infeasible. Here we simply truncate the high-order terms, and obtain the following linear form:

$$\mathtt{sim}(\mathbf{b}_i, \mathbf{t}_j) \approx 1 + \mathbf{b}_i^T \cdot \mathbf{t}_j/\tau, \tag{6}$$

More importantly, l_2 normalization enforces that $-1 \le \mathbf{b}_i^T \cdot \mathbf{t}_j \le 1$, and the temperature is generally chosen with $\tau \ge 1$ [15], our $\mathtt{sim}(\mathbf{b}_i, \mathbf{t}_j)$ is thus subject to non-negative constraint. To show the benefits of Taylor series, we substitute Eq. (6) into Eq. (4) and use the fact that $(\mathbf{x}^T\mathbf{y})\mathbf{x} = (\mathbf{x}\mathbf{x}^T)\mathbf{y}$, the Eq. (4) can be further simplified as:

$$\begin{aligned} \bar{\mathbf{t}}_j^{(b\to t)} &= \frac{\sum_{i=1}^{R_b}(1 + \mathbf{b}_i^T \cdot \mathbf{t}_j/\tau)\mathbf{b}_i}{\sum_{i=1}^{R_b}(1 + \mathbf{b}_i^T \cdot \mathbf{t}_j/\tau)} \\ &= \frac{\tau \sum_{i=1}^{R_b} \mathbf{b}_i + \sum_{i=1}^{R_b}(\mathbf{b}_i\mathbf{b}_i^T)\mathbf{t}_j}{\tau R_b + \mathbf{t}_j^T \sum_{i=1}^{R_b} \mathbf{b}_i}, \end{aligned} \tag{7}$$

we further observe that the terms $\tau \sum_{i=1}^{R_b} \mathbf{b}_i$, $\sum_{i=1}^{R_b}(\mathbf{b}_i\mathbf{b}_i^T)$, τR_b, and $\sum_{i=1}^{R_b} \mathbf{b}_i$ are independent on index j, which can be computed once and reused for every query. As a result, our Eq. (7) can lightly get rid of the exponential function in the vanilla softmax attention (e.g., Eq. (3)) without any additional cost. This yields a much simpler attention structure and achieves linear time and space complexity with respect to R for all queries.

To this end, we can efficiently compute the aggregated messages $\bar{\mathbf{T}}^{(b\to t)} \in \mathbb{R}^{d\times R_t}$ for the path bottom-to-top. Likewise, we can obtain the aggregated messages $\bar{\mathbf{B}}^{(t\to b)} \in \mathbb{R}^{d\times R_b}$, $\bar{\mathbf{B}}^{(s\to b)} \in \mathbb{R}^{d\times R_b}$, and $\bar{\mathbf{S}}^{(b\to s)} \in \mathbb{R}^{d\times R_s}$ for the paths: top-to-bottom, shoe-to-bottom, and bottom-to-shoe, respectively.

3.3.2 ***Visual Fusion Gate.*** In general, the aggregated messages (e.g., $\bar{\mathbf{T}}^{(b\to t)}$) can contain irrelevant or negative (mismatched) signals *w.r.t* original features (e.g., $\mathbf{T}$), which need to be filtered out during pairwise matching. For example, a "blue jean" should pass more visual signals to a "white t-shirt", but less signals to a "red suit jacket".

To better align knowledge between two modules, we further adopt gated mechanisms to allow the model to control what information should be propagated across different modalities [37]. Consequently, the proposed model is able to fuse two objects to a large extent for matched regions, and suppress the fusion for mismatched regions. We next introduce how to fuse the bottom-to-top messages $\bar{\mathbf{T}}^{(b\to t)} \in \mathbb{R}^{d\times R_t}$ and the original top features $\mathbf{T} \in \mathbb{R}^{d\times R_t}$ in details.

Recall that $\mathbf{t}_i$ is the original feature of top, and $\bar{\mathbf{t}}_i^{(b\to t)}$ is the message to be passed from bottom to top, we design a learnable gate for each region channel as:

$$\mathbf{g}_i^{(b\to t)} = \sigma(\mathbf{t}_i \odot \bar{\mathbf{t}}_i^{(b\to t)}), \quad 1 \le i \le R_t, \tag{8}$$

where $\odot$ denotes the Hadamard product, $\sigma(\cdot)$ is the sigmoid function, and $\mathbf{g}_i^{(b\to t)} \in \mathbb{R}^d$ is the gate for i-th pair of regions, whose elements are normalized between 0 (no fusion) and 1 (complete fusion). Thereafter, the region-level gates can be represented as $\mathbf{G}^{(b\to t)} = [\mathbf{g}_1, \ldots, \mathbf{g}_{R_t}] \in \mathbb{R}^{d\times R_t}$, which can help to filter trivial messages.

Besides, to preserve the original features $\mathbf{T}$ that should not be intensively fused by its neighbors (e.g., bottom), a residual connection is also applied:

$$\tilde{\mathbf{T}} = \mathbf{T} + \mathcal{F}^{(b\to t)}(\mathbf{G}^{(b\to t)} \odot (\mathbf{T} \oplus \bar{\mathbf{T}}^{(b\to t)})), \tag{9}$$

where $\mathcal{F}^{(b\to t)}(\cdot)$ is an one-layer feed forward network with ReLU activation, $\oplus$ is the element-wise summation, and $\tilde{\mathbf{T}} \in \mathbb{R}^{d\times R_t}$ is the fused region-wise features for top clothing. The residual connection also helps gradients flow through the layers to improve training stability.

Similarly, we can design the gated units for bottom and shoe as:

$$\begin{aligned} \tilde{\mathbf{B}} &= \mathbf{B} + \mathcal{F}^{(t\to b)}(\mathbf{G}^{(t\to b)} \odot (\mathbf{B} \oplus \bar{\mathbf{B}}^{(t\to b)})) \\ &\quad + \mathcal{F}^{(s\to b)}(\mathbf{G}^{(s\to b)} \odot (\mathbf{B} \oplus \bar{\mathbf{B}}^{(s\to b)})), \\ \tilde{\mathbf{S}} &= \mathbf{S} + \mathcal{F}^{(b\to s)}(\mathbf{G}^{(b\to s)} \odot (\mathbf{S} \oplus \bar{\mathbf{S}}^{(b\to s)})), \end{aligned} \tag{10}$$

where $\tilde{\mathbf{B}} \in \mathbb{R}^{d\times R_b}, \tilde{\mathbf{S}} \in \mathbb{R}^{d\times R_s}$ are the fused region-wise features for bottom and shoe, respectively. Note that bottoms can receive both cross-modal messages from tops and shoes concurrently, according the path: *top* $\leftrightarrow$ *bottom* $\leftrightarrow$ *shoe*.

Finally, pooling operations can be used to summarize the region-wise features into a compact vector for a whole image [23]. To be specific, given fused features $\tilde{\mathbf{T}}$, $\tilde{\mathbf{B}}$, and $\tilde{\mathbf{S}}$, we adopt a mean-pooling operation (e.g., average over regions) with three subsequent fully-connected layers $g(\Theta; \cdot)$ to obtain the final vectors:

$$\begin{aligned} \hat{\mathbf{t}} &= g(\tilde{\Theta}_t; \mathtt{avg\_pool}(\tilde{\mathbf{T}})), \quad \hat{\mathbf{b}} = g(\tilde{\Theta}_b; \mathtt{avg\_pool}(\tilde{\mathbf{B}})), \\ \hat{\mathbf{s}} &= g(\tilde{\Theta}_s; \mathtt{avg\_pool}(\tilde{\mathbf{S}})), \end{aligned} \tag{11}$$

where $\hat{\mathbf{t}} \in \mathbb{R}^{\hat{d}}$, $\hat{\mathbf{b}} \in \mathbb{R}^{\hat{d}}$, and $\hat{\mathbf{s}} \in \mathbb{R}^{\hat{d}}$ are unified representations for the whole images of top, bottom, and shoe, respectively. To this end, the unified vectors contain both the original features as well as the messages passing from their neighbors, which is capable of capturing more informative feature interplay, such as local compatibility. In addition, Eq. (11) aims to embed the visual features into a $\hat{d}$-dimensional joint space, which can be used to measure the full-body style compatibility and fashionability via Tensor Network.

3.4 Tensor Network

3.4.1 ***Wide&Deep Learning.*** Here we present how to measure the global compatibility among tops, bottoms, and shoes. Specifically, given a top $I_t^{(p)}$, a bottom $I_b^{(q)}$, and a shoe $I_s^{(r)}$, we can obtain their unified embeddings $\hat{\mathbf{t}}_p$, $\hat{\mathbf{b}}_q$, and $\hat{\mathbf{s}}_r$ via Eq. (11). To estimate their visual compatibility score, we propose to use a Wide & Deep Tensor Network as:

$$\hat{\mathcal{X}}_{pqr} = \sigma\left(\mathbf{W} \times \begin{bmatrix} \hat{\mathbf{t}}_p \odot \hat{\mathbf{b}}_q \odot \hat{\mathbf{s}}_r \\ \texttt{MLP}(\hat{\mathbf{t}}_p, \hat{\mathbf{b}}_q, \hat{\mathbf{s}}_r) \end{bmatrix}\right), \tag{12}$$

where $\sigma(\cdot)$ is the sigmoid function, $[\cdot]$ denotes the vector concatenations, and $\mathbf{W}$ is used to project the concatenated vector to a final score. The term $[\hat{\mathbf{t}}_p \odot \hat{\mathbf{b}}_q \odot \hat{\mathbf{s}}_r]$ is a wide network that pools a set of embeddings to one vector, and more importantly, it does not introduce extra model parameter. On the other hand, $\texttt{MLP}(\hat{\mathbf{t}}_p, \hat{\mathbf{b}}_q, \hat{\mathbf{s}}_r)$ is a deep network that extracts a vector from outfit embeddings by applying perceptron layers hierarchically. Joint training wide and deep networks enables the model to obtain better feature interactions among the unified embeddings for an outfit.

Compared to multi-linear tensor models, e.g., CANDECOMP/PARAFAC (CP) or Tucker [21], our TensorNet allows to learn non-linear feature interactions for multi-aspect tensor data. This greatly facilitates to improve the expressive power of tensor machines. Essentially, our TensorNet is a natural extension to the well-known Wide&Deep learning frameworks [7, 9, 14], by generalizing matrix models to high-order tensor models. These hybrid architectures combine the benefits of memorization generalization that can provide more insight for improving the system performance.

3.4.2 ***K-pair Loss Objective.*** To learn model parameters, we opt for the margin-based ranking loss function [5], which encourages the discrimination between positive triplets and negative triplets. That is to minimize

$$\mathcal{L}(\Theta) = \sum_{(p,q,r)\in\mathcal{T}} [1 + f(p',q',r') - f(p,q,r)]_+,$$

where $[x]_+ = \max(x, 0)$, $f(\cdot)$ and Θ are the predictive function and model parameters in Eq. (12), respectively. $\mathcal{T}$ denotes the training set, in which each triplet (p, q, r) is a positive sample (e.g., $\mathcal{X}_{pqr} = 1$), and (p', q', r') is a negative triplet corresponding to the positive (p, q, r), which can be randomly generated such that $\mathcal{X}_{p'q'r'} = 0$.

Although the above contrastive loss can learn the model parameters efficiently, it often suffers from slow convergence and poor local optima [29]. These issues arise from the fact that the contrastive loss only compares a positive sample with one negative sample at a single update of the model parameters. As a result, the embeddings of a positive sample is only guaranteed to be far from the selected negative sample, but not necessarily the others.

Inspired by K-pair loss [29], we seek to identify a positive example from multiple negative samples. Given one positive sample $(p, q, r) \in \mathcal{T}$ and its K negative samples $\{(p'_k, q'_k, r'_k)\}_{k=1}^K$, the K-pair contrastive loss is defined as

$$\mathcal{L}_K(\Theta) = -\sum_{(p,q,r)\in\mathcal{T}} \log \frac{\exp(f(p,q,r))}{\exp(f(p,q,r)) + \sum_{k=1}^K \exp(f(p'_k, q'_k, r'_k))} \tag{13}$$

The K-pair loss recruits multiple negative samples for each update, which accelerates the convergence and provides better optima. Specially, when $K = 1$, the Eq. (13) becomes $\mathcal{L}_1(\Theta) = \sum_{(p,q,r)\in\mathcal{T}} \log(1 + \exp(f(p', q', r') - f(p, q, r)))$, which is optimally equivalent to the margin-based ranking loss [29].

3.4.3 ***Computational Complexity.*** The time complexity of TensorNet, for each training triplet, mainly comes from three modules. For cross-attention message passing module, it takes $O(dR)$ to compute the pairwise messages, where d is the embedding dimension in Eq. (1) and $R = \max(R_t, R_b, R_s)$. For the visual gate units, the time cost of fusing features is $O(d^2R)$. For wide&deep tensor layer, the cost of wide component is $O(\hat{d})$ for its vector pooling. The matrix multiplication in deep component is $O(\sum_{h=1}^H \hat{d}_h \hat{d}_{h-1})$, where H is the number of hidden layers in MLP and $\hat{d}_h$ is the size of the h-th hidden layer, with $\hat{d}_0 = \hat{d}$. Although the K-pair loss takes more time than the margin-based ranking loss, an effective batch construction can be used to identify one positive sample from multiple negative samples simultaneously [29]. Therefore, the overall time complexity of TensorNet is $O(d^2R + \sum_{h=1}^H \hat{d}_h \hat{d}_{h-1})$ in total.

Table 1: Dataset statistics.

Dataset	# Top	# Bottom	# Shoe	# Interactions
Polyvore	15,806	14,869	15,706	28,360
iFashion	6,353	5,756	5,968	24,802

4 EXPERIMENTS

We evaluate our proposed TensorNet on two real-world datasets: Polyvore and iFashion. We aim to investigate the following research questions:

- **RQ1:** How does the proposed TensorNet perform compared with state-of-the-art recommendation methods?
- **RQ2:** Is our linearized attention capable of achieving similar performance with existing Linear Transformers?
- **RQ3:** What is the contribution of various components in the TesnorNet architecture (i.e., cross-attention mechanism, visual gated unit, wide&deep modules, K-pair loss.)?

4.1 Experimental Setup

Datasets: We evaluate our proposed TensorNet on two fashion datasets: Polyvore[6] and iFashion [8]. For both datasets, we collect three high-level categories (i.e., top, bottom, and shoe) from an outfit. Specifically, we first determine the low-level categories of

[6]http://www.polyvore.com. The Polyvore dataset was collected before the shutdown of its website.

each item by its textual descriptions, and then create a low-to-high category mapping to map any item into its corresponding high-level category. The top category contains 15 low-level categories: *bandeau, blouse, button-down, button-up, cardigan, camisole, pullover, shirt, shrug, sweater, tank, tee, top, turtleneck, vest*; the bottom category contains 15 low-level categories: *bermuda, bottom, chino, jean, jogger, jumpsuit, kilt, legging, overall, pant, romper, skirt, short, skort, trouser* ; and the shoe category contains 19 low-level categories: *boot, brogue, clog, converse, espadrille, flat, footwear, gladiator, heel, loafer, mule, moccasin, pump, sandal, slingback, slipper, shoe, sneaker, wedge*. We only keep outfits that have items in all three categories. The final statistics of the two datasets, after being processed into outfit triplets, are shown in Table 1.

Baselines: We compare TensorNet with matrix/tensor completion methods. While there are well established traditional tensor models (e.g., CP, Tucker, and their variants [21]), they suffer from high memory requirements caused by the process of flattening the entire tensors in each iteration, which leads to limited scalability. In this work, we choose the methods that allow the implementations of mini-batch training. They are: 1) NeuMF [14]: a deep matrix method capturing non-linear interactions between users and items. 2) VBPR [13]: a Bayesian ranking method with visual signals. 3) PITF [28]: a linear tensor model for context-aware recommendation. 4) NTF [34]: a neural tensor model with feed-forward network. 5) DCFA [39], a tensor model with pre-trained image features. 6) CoSTCo [25]: a CNN-based tensor model for sparse data.

Parameter Settings: For matrix-based models NeuMF and VBPR, we project the top-bottom-shoe tensor into two bipartite matrices: top-bottom and bottom-shoe. The candidate triplets are then generated by jointly learning the two matrices. The parameters of the baselines are initialized as in the original papers and are then carefully tuned to achieve optimal performance for each dataset. Also, the methods VBPR, DCFA, and TensorNet require pretrained CNN features. For a fair comparison, we implement these methods using ResNet50 (pretrained on ImageNet) as the underlying network, where the output of layer `pool5` (size: 2048) is used for the visual vectors for VBPR and DCFA, and the output of layer `conv5_3` (size: $7 \times 7 \times 2048$) is used for region-wise feature maps for TensorNet, i.e., each image is divided into 7×7 regions.

For TensorNet, the embedding dimension d in Eq. (1) is searched among $\{32, 64, 128, 256\}$, and the size of last fully-connected layer in Eq. (11) is set to $\hat{d} = d/2$. For deep component $\texttt{MLP}(\cdot)$ in Eq. (12), we employ two hidden layers and each layer sequentially decreases to half size of its input. The number of negative samples is set as $K = 2$ in Eq. (13). We implement our TensorNet in the TensorFlow with Adam optimizer. The batch size and learning rate are tuned within $\{128, 256, 512, 1024\}$ and $\{0.0005, 0.001, 0.005\}$, respectively.

Evaluation Metrics: We randomly split the observed triplets into 80% training, 10% validation, and 10% test sets. The validation set is used for tuning hyper-parameters and the final performance is conducted on the test set. We choose two widely used metrics: Hit@n and NDCG@n [5, 14], to evaluate the performance. For evaluation, we follow the similar *fill-in-the-bank* procedures as in [5]. Specifically, we consider three fill-in-the-bank scenarios: top-fill, bottom-fill, and shoe-fill. For top-fill scenario, given each test triplet (p, q, r), the top p is replaced by p' so that the (p', q, r) is unobserved (e.g., $\mathcal{X}_{p'qr} = 0$). We compute the compatibility scores for these corrupted triplets as well as the true triplet. Based on the ranking results, Hit@n and NDCG@n are computed. The similar strategy is applied to the scenarios of bottom-fill (e.g., replacing bottom q by q', such that $\mathcal{X}_{pq'r} = 0$) and shoe-fill (e.g., replacing bottom r by r').

4.2 Fill-in-the-blank Performance Comparison (RQ1)

Here we compare the fill-in-the-blank performance with the baselines under different scenarios. Fig. 3 summarizes the overall performance for the two datasets in terms of Hit@n and NDCG@n, where n is set to $\{10, 20\}$. We omit the results for other settings of n, which have a similar trends in the experiments. As shown in Fig. 3, our proposed TensorNet consistently yields the best performance across all cases. In addition, we have the following observations:

- Compared with matrix-based models (NeuMF and VBPR), tensor-based methods on average achieve better performance. This is attributed to the stronger ability of tensors when exploiting inherent relationships among multi-modal data. In our case, simply projecting the $top \times bottom \times shoe$ tensor into multiple matrices may break the original multi-way structure of an outfit and weaken the dependencies among tops, bottoms, and shoes.
- Among tensor models, the approaches (e.g., DCFA and TensorNet) that incorporate visual signals perform much better than pure tensor factorization methods (e.g., PITF, NTF, and CoSTCo) that rely only on the triplet links. In building capsule wardrobes, one would not buy clothes without seeing their shapes, colors, styles, etc. The visual appearance of clothes thus plays an important role in compatibility. More importantly, integrating visual features helps alleviate cold-start issue, especially for sparse data like outfit dataset.
- As expected, TensorNet achieves the best performance over all baselines, showing an average improvement of 9.16% and 10.33% than the state-of-the-art DCFA tensor model in terms of Hit@n and NDCG@n ($n = \{10, 20\}$), respectively. The improvements of TensorNet mainly come from its core visual modules: cross-attention modules and visual gated units. As such, TensorNet seeks to locally match regions of interest among tops, bottoms, and shoes, resulting in more appealing style compatibility. In addition, TensorNet adopts the effectiveness of wide&deep learning strategies to capture fine-grained global compatibility for an outfit.

4.3 Linearized Attention Analysis (RQ2)

Traditional softmax attention has been the computational bottleneck for many machine learning tasks. In this work, we linearized the softmax attention by using Talyor Series, resulting in better time and memory complexity. In this subsection, we further compare our linearized methods (e.g., Eq. (7)) with the vanilla softmax attention (e.g., Eq. (3)) and the Linear Transformer (e.g., Eq. (6)).

Table 2 shows the performance of different attention mechanism in terms of Hit@10 and NDCG@10, for Polyvore dataset. The similar results can be obtained for iFashion dataset and are omitted

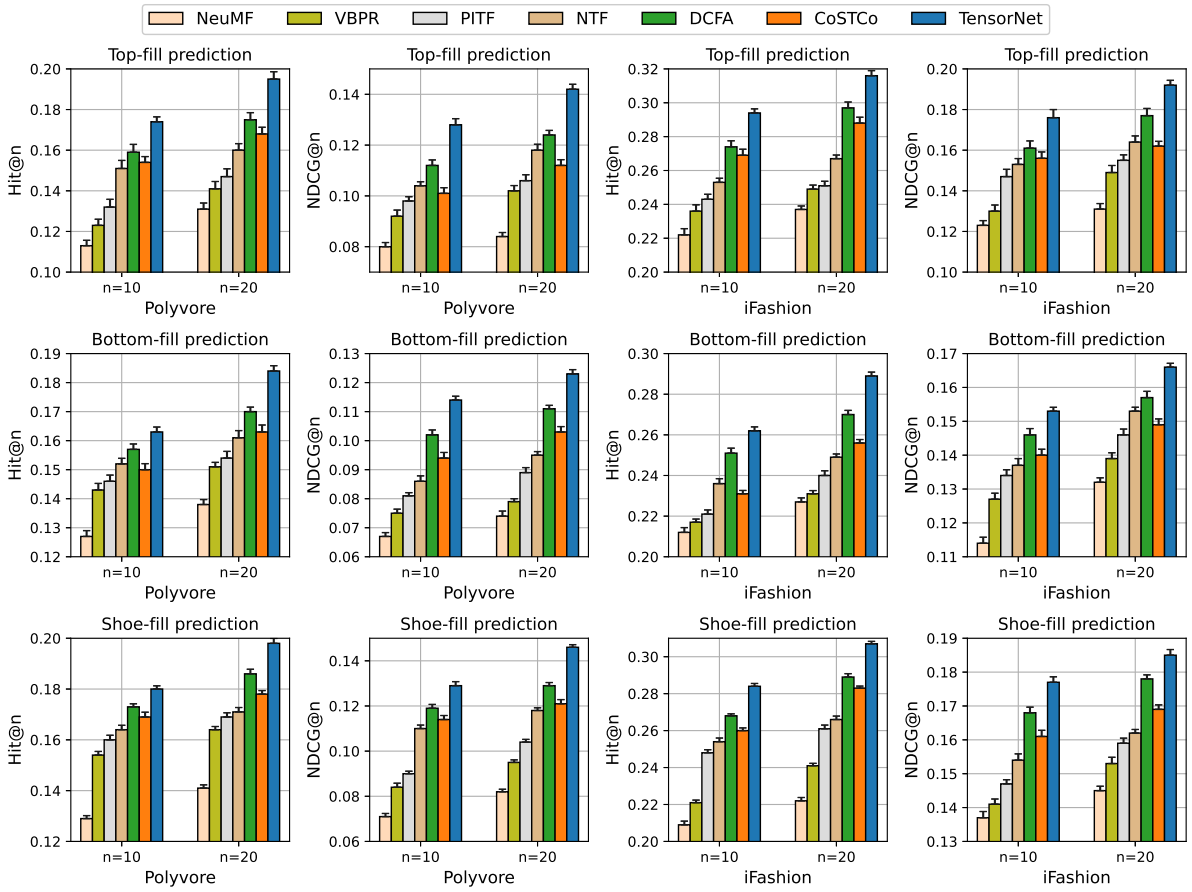

Figure 3: The fill-in-the-blank predictive results with error bars for different methods.

Table 2: Performance comparison for different attention mechanisms on Polyvore dataset. %Improv. denotes the relative improvements of our method over the Linear Transformer. The best results are highlighted in bold and the second best ones are underlined.

Polyvore	Top-fill scenario		Bottom-fill scenario		Shoe-fill scenario	
Metric	Hit@10	NDCG@10	Hit@10	NDCG@10	Hit@10	NDCG@10
Softmax Attention	**0.177**	**0.132**	**0.165**	**0.117**	**0.182**	**0.133**
Linear Transformer	0.169	0.125	0.156	0.109	0.175	0.124
Ours	<u>0.174</u>	<u>0.128</u>	<u>0.163</u>	<u>0.114</u>	<u>0.180</u>	<u>0.129</u>
%Improv.	2.96%	2.40%	4.49%	4.59%	2.86%	4.03%

here. Indeed, the vanilla softmax attention achieves the best performance, but with quadratic complexity for both running time and memory space. Compared to the Linear Transformer, our linearized attention method gains average improvements of 3.43% on Hit@10 and 3.68% on NDCG@10. Although the overall improvements are mild, our method has a straightforward mathematical explanation since the Taylor series expansion for exponential function is well-studied. Overall, the experimental results show that our proposed attention mechanism has a good trade-off between the effectiveness and complexity.

Fig. 4 also shows the top-3 pairwise salient matching patterns from our cross-attention maps. As mentioned earlier, cross-attention maps can be used to measure the region-to-region compatibility among tops, bottoms, and shoes. As one can see from the outfit on the top row, the portion that contributes to the matching between the pink sweater and the wide-leg pants are their stripe designs, and the matching between the pants and the pointy-toe shoes are the pointy shapes in both items. For the outfit in the bottom row, the matching between the shirt and the boot-cut jeans are their button designs, and the matching between the jeans and the booties

Figure 4: Top-3 pairwise salient matching patterns based on our cross-attention maps.

Table 3: Ablation analysis of our TensorNet. '↓' denotes a severe performance drop. H and N are short for Hit and NDCG.

Polyvore	Top-fill scenario			
Metric	H@10	N@10	H@20	N@20
TensorNet	0.174	0.128	0.195	0.142
TensorNet-wide	0.170	0.123	0.189	0.137
TensorNet-deep	0.172	0.125	0.192	0.140
Del gate units	0.164	0.122	0.186	0.135
Del cross-attention	0.161↓	0.119↓	0.181↓	0.133↓

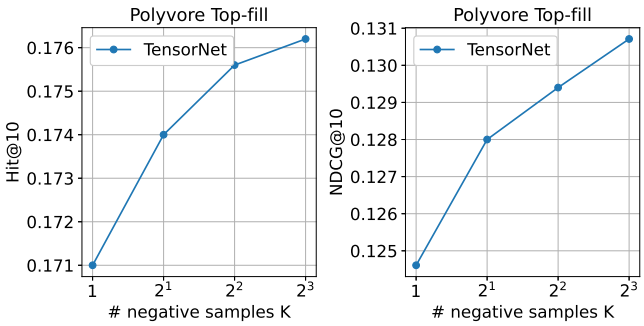

Figure 5: The impact of training multiple negative triplets $w.r.t$ one positive triplet simultaneously.

is the boot-cut silhouette and the high heels. These submodularity matching patterns can be further evaluated by fashion experts to see whether such region-to-region compatibility is aesthetically meaningful. In summary, our cross-attention map offers a unique tool for data-driven fashion advice in real-world applications.

4.4 Study of TensorNet (RQ3)

We further investigate the influence of each module in TensorNet via ablation studies. For each variant, we simply remove one network module and compare their performance with the default TensorNet.

Table 3 shows the performance of the four variants on Polyvore dataset. Our results are summarized as follows: 1) *TensorNet-wide*: this variant only train with wide component in Eq. (12), which decreases the performance, verifying the effectiveness of the MLP in modeling non-linear feature interactions; 2) *TensorNet-deep*: removing the wide component also hurts the system performance. Presumably this is because our wide network is able to preserve low-level features, which is important in the wide&deep learning; 3) *Delete gate units*: we find that our visual gate units can filter some irrelevant signals between two items, providing better results; 4) *Delete cross-attention module*: not surprisingly, removing cross-attention network significantly decreases the overall performance. This implies that the region-region matching is crucial to the outfit compatibility.

Fig. 5 also shows the impact of training multiple negative triplets $w.r.t$ one positive triplet simultaneously, where $K = \{1, 2, 4, 8\}$. We observe that our model benefits from a larger K, indicating the effectiveness of K-pair loss function in the training. Nevertheless, larger K leads to more running time. We found $K = 2$ is reasonable setting in our cases.

5 CONCLUSION

In this paper, we proposed TensorNet, a capsule wardrobe recommendation system that considers core pieces in fashion outfits, learns both the global and local compatibility between items by extracting regional feature maps, and leverages cross-attention message passing to pass visual signals between items that are closer to each other, while filtering out unwanted visual signals through visual gated units. In addition, we also design a linearized attention mechanism that learns the weights of regions in a scalable fashion. Our proposed TensorNet was evaluated on two large outfit datasets: Polyvore and iFashion. Both our quantitative and qualitative evaluations show that TensorNet outperforms other comparing methods by a large margin, and provides great explainability to capsule wardrobe generation.

REFERENCES

[1] Kenan E Ak, Ashraf A Kassim, Joo Hwee Lim, and Jo Yew Tham. 2018. Learning attribute representations with localization for flexible fashion search. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 7708–7717.
[2] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2018. Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 6077–6086.
[3] Ada S Ballin. 1885. *The science of dress in theory and practice*. Sampson, Low, Marston, Searle & Rivington.
[4] Irwan Bello. 2021. LambdaNetworks: Modeling long-range Interactions without Attention. In *International Conference on Learning Representations*.
[5] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Neural Information Processing Systems*. 1–9.
[6] Huiyuan Chen and Jing Li. 2019. Adversarial tensor factorization for context-aware recommendation. In *Proceedings of the 13th ACM Conference on Recommender Systems*. 363–367.
[7] Huiyuan Chen and Jing Li. 2020. Neural Tensor Model for Learning Multi-Aspect Factors in Recommender Systems.. In *IJCAI*. 2449–2455.
[8] Wen Chen, Pipei Huang, Jiaming Xu, Xin Guo, Cheng Guo, Fei Sun, Chao Li, Andreas Pfadler, Huan Zhao, and Binqiang Zhao. 2019. POG: personalized outfit generation for fashion recommendation at Alibaba iFashion. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*. 2662–2670.
[9] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, et al.

2016. Wide & deep learning for recommender systems. In *Proceedings of the 1st workshop on deep learning for recommender systems*. 7–10.

[10] Krzysztof Marcin Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamas Sarlos, Peter Hawkins, Jared Quincy Davis, Afroz Mohiuddin, Lukasz Kaiser, David Benjamin Belanger, Lucy J Colwell, and Adrian Weller. 2021. Rethinking Attention with Performers. In *International Conference on Learning Representations*.

[11] Xintong Han, Zuxuan Wu, Yu-Gang Jiang, and Larry S Davis. 2017. Learning fashion compatibility with bidirectional lstms. In *Proceedings of the 25th ACM international conference on Multimedia*. 1078–1086.

[12] Kaiming He, Xiangyu 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*. 770–778.

[13] Ruining He and Julian McAuley. 2016. VBPR: visual bayesian personalized ranking from implicit feedback. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 30.

[14] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In *Proceedings of the 26th international conference on world wide web*. 173–182.

[15] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *NeurIPS Deep Learning and Representation Learning Workshop* (2015).

[16] Ruibing Hou, Hong Chang, Bingpeng MA, Shiguang Shan, and Xilin Chen. 2019. Cross Attention Network for Few-shot Classification. In *Advances in Neural Information Processing Systems*.

[17] Wei-Lin Hsiao and Kristen Grauman. 2018. Creating capsule wardrobes from fashion images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 7161–7170.

[18] Yang Hu, Xi Yi, and Larry S Davis. 2015. Collaborative fashion recommendation: A functional tensor factorization approach. In *Proceedings of the 23rd ACM international conference on Multimedia*. 129–138.

[19] Alexandros Karatzoglou, Xavier Amatriain, Linas Baltrunas, and Nuria Oliver. 2010. Multiverse recommendation: n-dimensional tensor factorization for context-aware collaborative filtering. In *Proceedings of the fourth ACM conference on Recommender systems*. 79–86.

[20] Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. 2020. Transformers are rnns: Fast autoregressive transformers with linear attention. In *International Conference on Machine Learning*. 5156–5165.

[21] Tamara G Kolda and Brett W Bader. 2009. Tensor decompositions and applications. *SIAM review* 51, 3 (2009), 455–500.

[22] Zhanghui Kuang, Yiming Gao, Guanbin Li, Ping Luo, Yimin Chen, Liang Lin, and Wayne Zhang. 2019. Fashion retrieval via graph reasoning networks on a similarity pyramid. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 3066–3075.

[23] Kuang-Huei Lee, Xi Chen, Gang Hua, Houdong Hu, and Xiaodong He. 2018. Stacked cross attention for image-text matching. In *Proceedings of the European Conference on Computer Vision*. 201–216.

[24] Yusan Lin, Maryam Moosaei, and Hao Yang. 2020. OutfitNet: Fashion outfit recommendation with attention-based multiple instance learning. In *Proceedings of The Web Conference 2020*. 77–87.

[25] Hanpeng Liu, Yaguang Li, Michael Tsang, and Yan Liu. 2019. CoSTCo: A neural tensor completion model for sparse tensors. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 324–334.

[26] Hao Peng, Nikolaos Pappas, Dani Yogatama, Roy Schwartz, Noah Smith, and Lingpeng Kong. 2021. Random Feature Attention. In *International Conference on Learning Representations*.

[27] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In *Advances in Neural Information Processing Systems*.

[28] Steffen Rendle and Lars Schmidt-Thieme. 2010. Pairwise interaction tensor factorization for personalized tag recommendation. In *Proceedings of the third ACM international conference on Web search and data mining*. 81–90.

[29] Kihyuk Sohn. 2016. Improved deep metric learning with multi-class n-pair loss objective. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*. 1857–1865.

[30] Mariya I Vasileva, Bryan A Plummer, Krishna Dusad, Shreya Rajpal, Ranjitha Kumar, and David Forsyth. 2018. Learning type-aware embeddings for fashion compatibility. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 390–405.

[31] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems*.

[32] Zihao Wang, Xihui Liu, Hongsheng Li, Lu Sheng, Junjie Yan, Xiaogang Wang, and Jing Shao. 2019. Camp: Cross-modal adaptive message passing for text-image retrieval. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 5764–5773.

[33] Xi Wei, Tianzhu Zhang, Yan Li, Yongdong Zhang, and Feng Wu. 2020. Multi-modality cross attention network for image and sentence matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10941–10950.

[34] Xian Wu, Baoxu Shi, Yuxiao Dong, Chao Huang, and Nitesh V Chawla. 2019. Neural tensor factorization for temporal interaction learning. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*. 537–545.

[35] Yunyang Xiong, Zhanpeng Zeng, Rudrasis Chakraborty, Mingxing Tan, Glenn Fung, Yin Li, and Vikas Singh. 2021. Nyströmformer: A Nyström-based Algorithm for Approximating Self-Attention. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

[36] Zenglin Xu, Feng Yan, and Yuan Alan Qi. 2012. Infinite Tucker Decomposition: Nonparametric Bayesian Models for Multiway Data Analysis. In *International Conference on Machine Learning*.

[37] Linwei Ye, Mrigank Rochan, Zhi Liu, and Yang Wang. 2019. Cross-modal self-attention network for referring image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10502–10511.

[38] Ruiping Yin, Kan Li, Jie Lu, and Guangquan Zhang. 2019. Enhancing fashion recommendation with visual compatibility relationship. In *The World Wide Web Conference*. 3434–3440.

[39] Wenhui Yu, Huidi Zhang, Xiangnan He, Xu Chen, Li Xiong, and Zheng Qin. 2018. Aesthetic-based clothing recommendation. In *Proceedings of the 2018 world wide web conference*. 649–658.

[40] Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, and Amr Ahmed. 2020. Big Bird: Transformers for Longer Sequences. In *Advances in Neural Information Processing Systems*.