PERSONAL STYLIST: AN INTELLIGENT EVALUATION MODEL FOR PERSONALIZED STYLING ADVICE

KAICHENG PANG

PhD

The Hong Kong Polytechnic University
2024

The Hong Kong Polytechnic University
School of Fashion and Textiles

Personal Stylist: An Intelligent Evaluation Model for Personalized Styling Advice

Kaicheng PANG

A thesis submitted in partial fulfilment of the
requirements for the degree
of
Doctor of Philosophy

August 2023

CERTIFICATE OF ORIGINALITY

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgement has been made in the text.

_____ (Signed)

PANG Kaicheng

_____ (Name of student)

Abstract

An intelligent personal stylist, designed to offer personalized fashion recommendations, is becoming increasingly important in the fashion industry. Meeting the crucial requirements of **compatibility** and **personalization**, this technology aims to provide users with aesthetically pleasing and tailored fashion outfits. However, existing systems face limitations when deployed in real-world applications, prompting the need for further advancements in this field.

The existing literature exhibits three primary limitations: 1). Most existing fashion compatibility models overlook hierarchical relationships among fashion elements and need more explanatory capabilities; 2). Due to the lack of professional fashion knowledge in current evaluation datasets, the assessment of fashion compatibility models is limited to accuracy rather than aesthetic ability; 3). Existing research on personalized fashion recommendation mainly prioritizes user preferences and social media data, neglecting to incorporate diverse appearances of customers in the recommendation process.

This thesis aims to develop an intelligent personal stylist to overcome the above limitations. Firstly, a Hierarchical Outfit Network is proposed, featuring a multi-layered structure that captures relations among attributes, items, and outfits. The model learns outfit representation in a bottom-up manner, utilizing the attention mechanism to model feature relationships at each level. A gradient penalty loss is also employed to learn the underlying reasons behind the compatibility prediction.

Secondly, a novel evaluation dataset, Aesthetic 100 (A100), is developed to assess the aesthetic capabilities of fashion compatibility models. A100 demonstrates three desirable qualities: 1). Completeness, covering two types of standards in the fashion aesthetic system through two independent aesthetic tests;

2) Reliability, being independent of training data and consistent with major indicators; 3) Explainability, identifying essential fashion aesthetic characteristics to evaluate model performance in more detailed dimensions.

Thirdly, a new fashion cognition modeling task is introduced to investigate the relationship between outfits and an individual's physical attributes. A new dataset is constructed, consisting of 29,352 annotated outfits that indicate physical attribute compatibility. Moreover, a Fashion Convolutional Network is proposed to solve the task, comprising an outfit encoder module that encodes fashion attribute features into an outfit embedding using convolutional layers of various window sizes and a multi-label graph convolutional network module that captures label correlations to learn classifiers for physical attributes. The compatibility score is obtained by applying the classifiers to the outfit embedding.

Fourthly, a novel framework, Body-shape-Aware Network, is developed to enhance body-aware recommendations. This network utilizes visual and anthropometric features from a large-scale body shape dataset to represent body shapes. It also incorporates try-on images generated by the proposed Multi-layer Virtual Try-on System to represent outfits. The cross-model attention mechanism is leveraged to provide attribute-level explanations.

The contributions of this thesis lie in developing and advancing intelligent fashion recommendation systems. The proposed solutions address the challenges of fashion compatibility and cognition modeling, providing practical and effective tools for the fashion industry to enhance customer experiences. The research also reveals its limitations and provides insights for further exploration.

Publications arising from the thesis

1. Kaicheng Pang, Xingxing Zou, and Waikeung Wong. Modeling fashion compatibility with explanation by using bidirectional lstm. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 3894–3898, June 2021

2. Kaicheng Pang, Xingxing Zou, and Waikeung Wong. Dress well via fashion cognitive learning. In *33rd British Machine Vision Conference 2022, BMVC 2022, London, UK, November 21-24, 2022*, page 251. BMVA Press, 2022

3. Kaicheng Pang, Xingxing Zou, Fangjian Liao, and Waikeung Wong. M-vton: Multi-layer virtual try-on system. *Design and Semantics of Form and Movement*, 266, 2023

4. Kaicheng Pang, Xingxing Zou, and Waikeung Wong. Towards intelligent online cross-selling. *Expert Systems with Applications*, 2022 (Under review)

5. Kaicheng Pang, Xingxing Zou, and Waikeung Wong. Learning visual body-shape-aware embeddings for fashion compatibility. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2024 (Accepted)

6. Xingxing Zou, Kaicheng Pang, Wen Zhang, and Waikeung Wong. How good is aesthetic ability of a fashion model? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21200–21209, 2022

Acknowledgements

I would like to express my deepest gratitude to my supervisor, Prof. Wai Keung Wong, for his professional guidance, invaluable insights, and continuous support throughout my doctoral journey. His expertise, dedication, and encouragement have played a pivotal role in shaping this research and have pushed me to strive for excellence.

I extend my heartfelt appreciation to the faculty and staff of the School of Fashion and Textiles for providing a nurturing academic environment and valuable resources that have been indispensable for my research endeavors.

I am sincerely grateful to my colleagues and fellow researchers, including Dr. Xingxing Zou, Dr. Dongmei Mo, Mr. Fangjian Liao, Miss Shumin Zhu, Mr. Sibo Wang, and other friends who have supported me throughout this journey. Their collaboration, insightful discussions, and shared experiences have been a constant source of inspiration and motivation.

Furthermore, I would like to express my heartfelt thanks to my family, especially my wife, Dr. Linli Zhang, without whom I would not have been able to embark on this PhD journey. Their enduring love, encouragement, and understanding have provided me with the strength and determination to overcome challenges. I am deeply grateful for their sacrifices and unwavering support throughout this endeavor.

Table of contents

Abstract **ii**

Publications arising from the thesis **iv**

Acknowledgements **v**

List of figures **xi**

List of tables **xiv**

1 Introduction **1**

1.1 Research Background 1

1.1.1 Fashion Compatibility Modeling (FCM) 5

1.1.2 Fashion Cognition Modelling 7

1.2 Research Objectives 9

1.3 Research Methodology 10

1.3.1 Modeling Fashion Compatibility 10

1.3.2 Modeling Fashion Cognition 11

1.4 Research Significance 13

1.5 Organization of the Thesis 16

2 Literature Review **18**

2.1 Fashion Compatibility Modeling 18

2.1.1 State-of-the-art Methods 19
2.1.2 Benchmark Datasets 22
2.2 Personalized Fashion Recommendation 23
2.2.1 State-of-the-art Methods 23
2.2.2 Benchmark Datasets 25
2.3 Evaluation Metrics 26
2.4 Chapter Summary 28

3 How Good Is Aesthetic Ability of a Fashion Model? 30
3.1 Introduction 30
3.2 Related Work 34
3.3 A100 Evaluation Protocol 35
3.3.1 Outfit Generation Principle 35
3.3.2 Liberalism Aesthetic Test (LAT) 37
3.3.3 Academicism Aesthetic Test (AAT) 42
3.4 Analysis 45
3.4.1 Analysis of Reliability 45
3.4.2 Analysis of Explainability 49
3.5 Chapter Summary 53

4 Hierarchical Outfit Network for Fashion Compatibility Learning 54
4.1 Introduction 54
4.2 Related Work 57
4.3 Hierarchical Outfit Network 58
4.3.1 Problem Formulation 59
4.3.2 Multi-head Attention 59

4.3.3 Attribute Level Network 60
4.3.4 Item Level Network 61
4.3.5 Outfit Level Network 62
4.3.6 Objective Function 62
4.4 Experiments 63
4.4.1 Experimental Settings 64
4.4.2 Compared Methods 65
4.4.3 Quantitative Results (RQ1) 67
4.4.4 Ablation Study (RQ2) 70
4.4.5 Empirical analysis on qualitative results (RQ3) 71
4.5 Application 74
4.5.1 Complementary Item Retrieval 74
4.5.2 Fashion Outfit Evaluation 75
4.5.3 Fashion Outfit Generation 76
4.6 Chapter Summary 76

5 Modeling Fashion Compatibility with Convincing Reasons 78
5.1 Introduction 78
5.2 Related Work 81
5.2.1 Explainable Fashion Compatibility Model 81
5.2.2 Long Short-term Memory (LSTM) networks 81
5.3 Approach 82
5.3.1 Feature Extraction Architecture 83
5.3.2 Bidirectional LSTM Architecture 84
5.3.3 Gradient Penalty Architecture 86
5.4 Experiments 88
5.4.1 Experimental Settings 88

5.4.2 Quantitative Analysis 89
5.4.3 Qualitative Analysis 90
5.4.4 Ablation study 93
5.5 Chapter Summary 94

6 Dress Well via Fashion Cognition Learning 96
6.1 Introduction 96
6.2 Related Work 99
6.3 O4U Dataset 100
6.4 Approach 101
6.4.1 Problem Formulation and Motivation 101
6.4.2 Outfit Encoder 103
6.4.3 Multi-label Graph Convolutional Networks 104
6.5 Experiments 106
6.5.1 Experimental Settings 106
6.5.2 Quantitative Results 107
6.5.3 Qualitative Results 110
6.5.4 Ablation Study 110
6.6 Chapter Summary 112

7 Learning Body-shape-Aware Embeddings for Fashion Compatibility 113
7.1 Introduction 113
7.2 Related Work 116
7.2.1 Body-shape-Aware Fashion Compatibility. 116
7.2.2 Outfit Representation 116
7.3 Body Shape Dataset 118

7.4 Multi-layer Virtual Try-on System 121
7.4.1 Keypoint Detection 121
7.4.2 Fashion Segmentation 122
7.4.3 Outfit Synthesis 124
7.5 Body-shape-Aware Network 125
7.5.1 Task Formulation 125
7.5.2 Body-shape Representation 125
7.5.3 Try-on Image Representation 128
7.5.4 Fashion Attributes Representation 128
7.5.5 Body-type-Aware Network Architecture 129
7.6 Experiments 131
7.6.1 Experimental Settings 131
7.6.2 Comparative Results (RQ1) 134
7.6.3 Ablation Study (RQ2) 136
7.6.4 Explainability Analysis (RQ3) 141
7.6.5 Perceptual Study (RQ4) 143
7.7 Chapter Summary 146

8 Conclusions and Suggestions for Future Research 147
8.1 Conclusions 147
8.2 Limitations 148
8.3 Suggestions for Future Research 150

References 151

A Limitations of Existing FITB Tests 176

B Qualitative Results of M-VTON on Type-aware Dataset 181

List of Figures

1.1 The market size of AI in fashion. 2
1.2 Three downstream tasks of fashion compatibility modeling. . . 5
1.3 Multiple relations between attribute, item, and outfit level . . 6
1.4 Two steps involved in fashion recommendations. 8
1.5 The structure of this thesis. 16

2.1 Statistics of evaluation indicators adopted in previous literature. 26

3.1 Illustration for the Aesthetic 100. 32
3.2 Examples of valid outfits. 36
3.3 An example question in LAT. 41
3.4 A FITB example in the Type-aware test set. 42
3.5 An example question in AAT examining *Color* index. 44
3.6 Experiments of finding the optimal model based on three validation sets. 47
3.7 Example questions of AAT examining the *Color* index. 51
3.8 Example question in LAT. 52

4.1 Relations of fashion outfits among attribute level, item level, and outfit level 55
4.2 Overview of the Proposed Hierarchical Outfit Network. 58
4.3 Qualitative results of different approaches. 72

4.4 Visualization results of implicit attributes. 73
4.5 Visualized examples of adopting HON on the Fashion outfit complementary item retrieval. 74
4.6 Visualized examples of adopting HON on the task of Fashion outfit evaluation and revision. 75
4.7 Visualized examples of adopting HON on the task of Fashion outfit generation and recommendation. 76

5.1 Pipeline of fashion compatibility network. 83
5.2 Overview of the Inter-factor Compatibility Network. 86
5.3 Qualitative analysis of the proposed approach on the expanded EVALUATION3 dataset. 90
5.4 A website demo application based on the proposed fashion compatibility model. 91
5.5 Effect on the number of LSTM layers. 93
5.6 Effect on the dimensions of the LSTM hidden layer. 94

6.1 Illustration of Fashion Cognition Learning task. 97
6.2 Number of examples for each physical label 101
6.3 An overview of the proposed Fashion Convolutional Network. . 102
6.4 Construction of adjacency matrix based on the conditional probability of *round* and *athletic*. 105
6.5 Qualitative results of baseline methods and the proposed FCN. 109

7.1 An example of the body-shape-aware fashion compatibility task. 114
7.2 Five common body shapes included in the newly introduced body shape dataset. 119
7.3 Overview of the multi-layer try-on system. 121

7.4 Illustration of segmentation model. 123
7.5 Overview of the proposed BA-Net. 126
7.6 Qualitative comparison among different methods. 135
7.7 Try-on results of outfits from the O4U dataset. 139
7.8 Visualization of different body features using t-SNE. 140
7.9 Visualization of attention maps computed in JEM. 142
7.10 The pipeline of a prototype for applying BA-Net in a real application. 145

A.1 FITB questions in the Maryland testing set. 177
A.2 FITB questions in the Type-aware testing set 179

B.1 Try-on results of outfits from the Type-aware Polyvore dataset. 182
B.2 Try-on results of outfits from the Type-aware Polyvore dataset. 183

List of Tables

2.1 Summary of the benchmark datasets for fashion compatibility modeling task. 22

2.2 Summary of the benchmark datasets for personalized recommendation. 25

3.1 Number of items for 20 clothing categories in mainstream datasets. 38

3.2 Specified factors for evaluating the model's ability of fashion aesthetic. 43

3.3 FITB evaluation results of four state-of-the-art methods. . . . 46

3.4 Results of optimal models based on different validation sets. . 48

3.5 Results of FHN, Bi-LSTMs, TSE, and SCE-Net evaluated on the AAT. 49

3.6 Performances comparison on the *Season* dimension. 50

4.1 Quantitative results on Maryland test set and Type-aware test set. 67

4.2 Quantitative results on A100. 68

4.3 Results of TSE and HON evaluated on the AAT. 69

4.4 Experimental results on different structures of network. . . . 70

4.5 Experimental results on different numbers of implicit attributes. 71

5.1 Fashion attributes of utilized in providing the prediction reason. 80

5.2 Comparison of different methods on the updated EVALUATION3 test set. 89

6.1 Details of personal physical features and sub-features. 100
6.2 Quantitative results on Body Shape attributes. 108
6.3 Quantitative results on the physical attributes excluding body shapes. 108
6.4 Quantitative results on main metrics. 108
6.5 Effect of filter region size. 110
6.6 Effect of numbers of kernels for each filter. 111
6.7 Effect of numbers of GCN layers. 111

7.1 Split details of segmentation dataset. 122
7.2 Quantitative comparison among different methods. 134
7.3 Ablation results on representation learning. 137
7.4 Comparison on variations of cross-modal attention. 138
7.5 Comparison of encoding outfits with and without try-on images. 138
7.6 Body shape classification accuracy comparing with available classifiers. 140
7.7 Perceptual results of the compatibility models. 144

Chapter 1

Introduction

1.1 Research Background

The fashion industry is one of the largest and most influential sectors globally, generating trillions of dollars in revenue each year [5]. Beyond its economic impact, fashion is an art form allowing individuals to express their identity. Online shopping platforms have revolutionized the industry, offering abundant choices and enabling the free exchange of fashion ideas [4]. However, the rise of e-commerce poses opportunities and challenges [39]. On the one hand, the fashion industry benefits from digital platforms, which enable global reach, expanded customer bases, and enhanced brand visibility. On the other hand, fashion companies face significant challenges in securing a competitive edge and driving sustainable growth by effectively utilizing emerging technologies and data-driven insights.

As a response to these challenges and opportunities, the integration of artificial intelligence (AI) technology into the fashion industry has emerged as a prominent trend in expanding digital fashion market. With the help of AI,

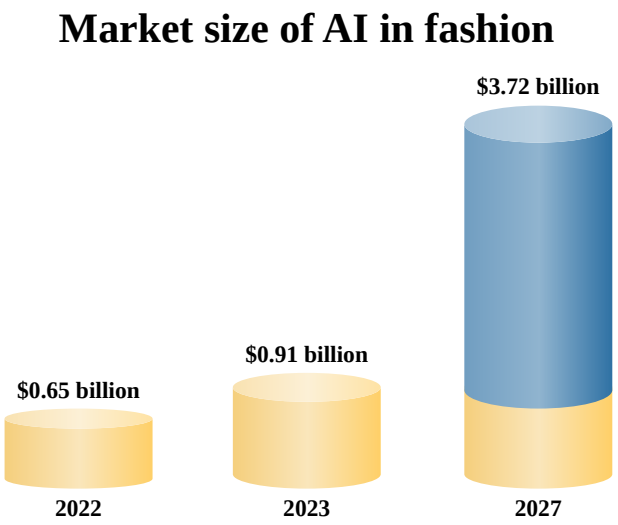


Figure 1.1 The market size of AI in fashion grow from $0.65 billion in 2022 to $0.91 billion in 2023. Moreover, it is projected to reach $3.72 billion by 2027, with a compound annual growth rate of 42.0%.

fashion companies can better understand customer preferences and provide personalized recommendations and marketing campaigns, ultimately increasing customer engagement and loyalty. AI also plays a crucial role in the design and manufacturing process by analyzing vast data on fashion trends, consumer tastes, and market demand, generating innovative and unique designs. Furthermore, AI optimizes the supply chain management system, reducing the cost of inventory management and logistics. These AI applications can potentially revolutionize the fashion industry, enhancing profitability and sustainability. According to the fashion global report [15], the AI in fashion industry witnessed significant growth in recent years. As illustrated in figure 1.1, the market size of AI in fashion has increased from $0.65 billion in 2022 to $0.91 billion in 2023, representing a 40.0% compound annual growth rate (CAGR). Moreover, it is projected that the AI fashion industry will continue to expand, reaching a market volume of $3.72 billion in 2027 at a higher CAGR of 42.0%.

The tremendous potential of AI in the fashion industry has attracted significant attention from researchers. Technically, intelligent fashion is a multidisciplinary field encompassing various research areas, including computer vision, machine learning, natural language processing, and human-computer interaction [13]. The current cutting-edge fashion research topics can be categorized into four main areas:

1). **Fashion detection** is extensively explored as it is a fundamental step for many fashion-related tasks aiming to detect fashion-related elements from the given image. It mainly consists of three tasks: (1) *fashion landmark detection* [79, 80, 148] whose objective is to forecast the locations of functional keypoints defined on clothing items; (2) *fashion parsing* [34, 109, 141] which segments the human image into pixel-level fashion elements, such as pants or dress; (3) *item retrieval* [65, 70, 74] which is an image-based fashion retrieval task aiming to match real-world fashion items with their corresponding online shopping images.

2). **Fashion analysis** aims to uncover personality traits for precise marketing and sociological analysis, demonstrating enormous potential in the fashion industry. It encompasses three primary tasks: (1) *attribute recognition* [16, 59, 125] which aims to identify fashion attributes from clothing items, and are usually modeled as a multi-label classification problem; (2) *style learning* [61, 86, 138] which refers to the process of understanding and identifying the discriminative features that distinguish different fashion styles and trend; (3) *popularity prediction* [6, 7, 33, 115] which tackles the problem of forecasting fashion popularity and future fashion trend based on the knowledge from fashion style learning.

3). **Fashion synthesis** aims to generate realistic-looking images that depict

a person in different makeup or clothing styles. It includes two main tasks: (1) *virtual try-on* [42, 149, 167] which enables the placement of desired clothing items onto the corresponding areas of a person in an image, creating a seamless visual representation; (2) *pose transformation* [106, 107, 117] which aims to generate pose-guided person images in different postures while preserving individual characteristics.

4). **Fashion recommendation** is an intelligent system recommending fashion products to customers. It can be hierarchically divided into two tasks: (1) *fashion compatibility modeling* [62, 89, 166] which evaluates the overall aesthetic quality of different types of clothing items collaborating to form fashionable outfits; (2) *personalized recommendation* which aims to provide personalized recommendations of fashionable outfits to customers, taking into account their physical attributes and preferences; (3) *explanable recommendation* [9, 129, 150] which enables recommendation system to provide users with a deeper understanding of why a specific item or outfit is recommended to them.

The summary above offers an extensive overview of the emerging application of AI within the realm of fashion. This thesis focuses on the **Fashion recommendation** problem, aiming to develop an intelligent personal stylist that offers customers personalized styling advice. However, achieving this goal faces significant challenges because it needs to fulfill two requirements: **compatibility** and **personalization**. The former involves the generation of compatible outfits, while the latter focuses on providing personalized outfit recommendations tailored to individuals. A brief literature review is presented in the subsequent subsection, focusing on the fundamental aspects of fashion recommendation systems. A more detailed review is provided in Chapter 2.

Figure 1.2 Three downstream tasks based on FCM. Task 1: Fashion outfit complementary item retrieval [71]. Task 2: Fashion outfit revision [169]. Task 3: Fashion outfit generation [18].

1.1.1 Fashion Compatibility Modeling (FCM)

The foundation of fashion recommendation lies in FCM, as illustrated in Figure 1.2. This task involves assessing the compatibility of multiple fashion items [19, 75, 83, 150, 160, 170]. In this domain, the development of compatibility models and the utilization of evaluation metrics are reviewed.

Fashion Compatibility Model

Predicting the compatibility of an outfit is complicated because it involves visual perception, texture, and trend, to name a few, and every factor is in the process of changing. Prior works conducted on compatibility prediction have

Figure 1.3 Multiple relations between attribute, item, and outfit level

explored numerous methods. The mainstream methods [17, 50, 134] adopt metric learning, where items of an outfit are transformed into embeddings. The embeddings of mismatching items are maximally separated, while the distances between matching items are minimized. Visileva *et al.* [134] proposed to learn an item embedding considering fashion categories. Cucurull *et al.* [17] utilized items' visual features and contextual information to forecast the compatibility of two items. Han *et al.* [41] treated individual items in an outfit as a sequence and employed bidirectional *Long Short-term Memory* networks to encode an outfit. Zheng *et al.* [166] proposed to evaluate compatibility from both the collocation and try-on perspectives. Other studies employed the *Conditional Random Field* [119] and clothing style modeling [2, 137] to estimate fashion compatibility. Despite the favorable outcomes achieved by these methods, they still exhibit certain areas for improvement:

(1). They overlook the intrinsic relationships among fashion data, limiting their evaluation performance. As shown in Figure 1.3, there are multi-relations between attributes, items, and outfits. Previous research only encoded outfit at item level [18, 71, 134] or attribute level [152, 169].

(2). The evaluation results of current compatibility models lack convincing

explanations. The explainability of compatibility models is essential for gaining users' trust in the recommendation system.

Evaluation Metrics

Apart from the accuracy of fashion compatibility models, evaluating these models presents additional challenges in fashion recommendation applications. The conventional approach to evaluation in the fashion industry often involves hiring professional stylists to assess the generated results, which not only incurs high costs but also proves inconvenient in terms of logistics and time. Furthermore, the current evaluation metrics employed in academic research primarily focus on measuring retrieval or ranking performance, such as the widely used *Area Under Curve* (AUC) metric. AUC has been applied to assess the compatibility score of item-level recommendations in various studies. Additionally, other commonly utilized metrics in recommendation tasks include *Normalized Discounted Cumulative Gain* (NDCG), *mean Average Precision* (mAP), and Recall. These metrics, although effective for ranked retrieval evaluation, fall short in capturing the aesthetic ability of compatibility models. They do not fully encompass essential fashion concepts such as compatibility, novelty, and beauty, which are crucial for evaluating the overall performance of these models. As a result, there is a clear need for more comprehensive and reliable evaluation metrics that can effectively measure the aesthetic qualities and fashion concepts addressed by compatibility models.

1.1.2 Fashion Cognition Modelling

Fashion cognition modelling is positioned at a higher level because it learns the relationship between the outfit and human physical information. As illus-

(a) Outfit compositions generated by fashion compatibility models

(b) Precise outfit recommendation with fashion cognition models

Figure 1.4 Two steps involved in fashion recommendations.

trated in Figure 1.4, there are two steps involved in fashion recommendation system. Figure 1.4 (a) depicts good mix-and-matches generated by the fashion compatibility model. However, as shown in Figure 1.4 (b), different customers have varied appearances, directly affecting whether an outfit is compatible with them. Taking the first outfit shown in Figure 1.4 (a) as an example, it consists of a long white dress unsuitable for the second customer since she is not so high enough to wear this long dress. Thus, even though this outfit is perfectly matched, it is not appropriate to recommend it to her. Otherwise, it will be resulting she is losing trust in the service provider. In other words, understanding the relationships between outfits and customers to achieve precise outfit recommendations is crucial.

Only a few works noticed the influence of personal information on the fashion recommendation problem. Most of them utilized user preference for personalized recommendations. Xu *et al.* [8] used user preference for individ-

ual items to satisfy the personalized recommendation requirements. Packer *et al.* [95] leveraged users' prior feedback to calculate their affinity matrix towards specific visual styles and attributes. Liu *et al.* [78] leveraged user reviews to assist recommender systems in predicting product ratings. Besides user preferences, social media posts are another commonly used information source [123, 165]. Zheng *et al.* [165] introduced a personalized fashion recommendation framework based on item-to-set metric learning based on social media data. Some studies investigated the relationships between the outfit and human shapes. Hsiao *et al.* [52] proposed a visual body-aware embedding framework to capture the item's association with diverse body shapes. Hidayati *et al.* [47, 48] modeled this relationship based on the full-body images of celebrities and their body measurement data collected from the internet. However, no prior approach systematically considered the varied appearance of individuals and then solved it from a comprehensive point of view.

1.2 Research Objectives

This study aims to develop an intelligent personal stylist targeting the fashion recommendations problem in the real application. Specifically, this research addresses the following objectives:

1. To construct a new fashion compatibility evaluation protocol that considers three aspects which are completeness, practicality, and reliability.

2. To propose novel fashion compatibility models that leverage the multi-relations among fashion elements and produce a convincing explanation for the evaluation result.

3. To construct a new dataset for the fashion cognition modeling task and propose a framework to accomplish this task.

4. To investigate the relationship between body shapes and outfits, utilizing visual images of body shapes and try-on appearances of outfits.

1.3 Research Methodology

1.3.1 Modeling Fashion Compatibility

The primary objective of modeling fashion compatibility is to build the understanding of how different clothing items complement each other. This study primarily focuses on three key aspects: constructing a comprehensive evaluation protocol, developing a hierarchical outfit network, and providing convincing explanations for compatibility prediction. These efforts collectively strive to enhance the model's fashion compatibility knowledge and facilitate the creation of visually harmonious outfits.

(1). Comprehensive Evaluation Protocol. To overcome the lack of a practical evaluation protocol for fashion compatibility models, a comprehensive evaluation protocol, named *Aesthetic 100* (A100), is constructed. A100 incorporates two types of fashion aesthetic standards: the *Bottom-up* and *Top-down* standards. Two tests, namely the Liberalism Aesthetic Test (LAT) and the Academicism Aesthetic Test (AAT), are designed to assess these standards, respectively. The LAT consists of randomly generated questions verified by fashion experts, with answers collected through a website questionnaire. The AAT examines the model's performance in six main fashion aspects, utilizing carefully created questions and options.

(2). Hierarchical Outfit Network. A hierarchical outfit network is proposed to improve the performance of the fashion compatibility model. This network leverages the multi-relations among fashion elements by incorporating three levels: attribute, item, and outfit. The network employs the multi-head attention mechanism [135] to model the internal relationships within each level. Fashion features are aggregated from the attribute level to the outfit level, resulting in an enhanced outfit representation. This hierarchical design enables the exploitation of tree-structured relations inherent in fashion data, leading to improved performance.

(3). Providing Convincing Explanation. The ability to offer explanations for the model's predictions is crucial in building user trust. Firstly, the EVALUATION3 dataset [169] is expanded, encompassing outfits with multiple items, attribute labels, compatibility judgments, and corresponding reasons. Secondly, based on this dataset, a new framework utilizing *bidirectional Long Short-term Memory* networks is proposed. During the training process, a gradient penalty regularization is employed to align the reasons predicted by the network with the reason labels in the dataset, thus enhancing the model's ability to provide explanations.

1.3.2 Modeling Fashion Cognition

The objective of modeling fashion cognition is to enable models to acquire knowledge pertaining to the interconnections between outfits and human physical attributes. This task is introduced and formulated as a multi-label classification problem. A comprehensive dataset comprising numerous outfits and corresponding physical labels is constructed. Moreover, two distinct models,

namely the Outfit Convolutional Network and the Body-shape-Aware Network, are proposed to address this task.

(1). Task Introduction, Dataset Construction, and Model Proposal.

Previous personalized fashion recommendation systems primarily focused on user preferences and neglected the physical attributes of customers. To address this shortcoming, the fashion cognition modeling task is introduced and formulated as a multi-label classification problem. A new dataset called *Outfit for You* (O4U) is constructed to facilitate task solving. The O4U dataset consists of 29,352 outfits and 82,677 annotations. Fashion experts are invited to annotate these outfits. Each outfit is assigned with two labels: one indicating its compatibility and the other identifying any incompatible physical attributes. Based on the O4U dataset, a *Fashion Convolutional Network* is proposed, which comprises an outfit encoder and a multi-label graph convolutional network (ML-GCN). The outfit encoder utilizes convolutional layers with diverse window sizes to encode fashion attribute features into an outfit embedding. The ML-GCN captures label correlations and learns classifiers for each physical attribute. The predicted compatibility score between the outfit and the physical label is obtained by multiplying the classifier features and outfit embedding.

(2). Enhancing Data Representation for Body Shape and Outfit.

A new framework named *Body-shape-Aware* network is proposed to enhance the model's fashion cognition in terms of body shapes. The body shapes are represented using visual and anthropometric features. A large-scale body shape dataset is created to provide the 3D models and images of body shapes. It contains 20,000 3D body models with varying body sizes generated using

the *Skinned Multi-Person Linear* (SMPL) [81] model. The dataset covers five common body shapes: the bottom hourglass, top hourglass, spoon, inverted triangle, and triangle.

The outfit is encoded by using its try-on appearance and fashion attributes. A Multi-layer Virtual Try-on System (M-VTON) is proposed to generate realistic try-on images given an outfit composed of various item images. The system extracts fashion keypoints using the pre-trained *ViPNAS* model. Scaling data and pixel locations of each item are calculated by aligning the extracted fashion keypoints with those on a mannequin image. The *Object-Contextual Representation* model is trained to segment clothing images into front and back pieces to achieve a multi-layered try-on effect. These segmented pieces are synthesized to create the try-on appearance according to the predefined try-on order. Outfit features are extracted from the obtained try-on image using the *Convolutional Neural Network* (CNN).

BA-Net is designed to provide the attribute-level explanation for its prediction by analyzing the attention maps computed by the cross-model attention mechanism. This approach enhances the interpretability of the evaluation process and enables users to understand the factors influencing the model's decisions.

1.4 Research Significance

(1). Introduction of a Comprehensive Evaluation Protocol.

This study makes a contribution by introducing a novel evaluation dataset named *Aesthetic 100*. Unlike existing datasets, A100 covers systematic aesthetic standards, comprehensively evaluating fashion compatibility quality. This pioneering work incorporates professional fashion domain knowledge, al-

lowing for more characteristic performance evaluation. Extensive experiments are conducted to demonstrate the proposed protocol's practicability and reliability.

(2). Enhancement of Aesthetic Ability and Explainability for Fashion Compatibility Model

This study proposes the *Hierarchical Outfit Network*, a novel approach that enhances performance by introducing implicit attributes and exploiting multi-relations among the attribute, item, and outfit levels. The HON surpasses state-of-the-art results on two widely used datasets and the Aesthetic 100 dataset, outperforming 14 baseline models. The study also proposes practical solutions for integrating the trained HON into real-world products, enabling effective cross-selling in online fashion platforms. In addition, employing a gradient penalty during model training enables the model to provide the reason aligned with experts' knowledge, enabling the system to deliver convincing explanations. These advancements in both performance and explainability foster greater user trust in the recommendation system, improving the overall user experience.

(3). Introduction of Fashion Cognition Modeling and O4U Dataset.

This study introduces the task of fashion cognition modeling and constructs the *Outfit for You* dataset. This work bridges the research gap in personalized fashion recommendation systems that previous works solely rely on user preferences while overlooking human physical attributes. The proposed *Fashion Convolutional Network*, trained on the O4U dataset, enables the fashion recommendation system to provide targeted clothing suggestions based on customers' physical information. By leveraging this innovative model,

the accuracy of fashion recommendations is improved, thereby increasing the potential sales and customer satisfaction in electronic retail.

(4). Enrichment of Methodology for Virtual Try-on System.

A *Multi-layer Virtual Try-on system* is presented that incorporates the technologies of fashion detection, alignment, and segmentation. In contrast to mainstream approaches relying on *Generative Adversarial Network* technology [35], the proposed try-on system exhibits notable advantages, including details preservation and high reliability. This system provides a realistic representation of outfits, enabling users to visualize how the clothing would look on them. From the perspective of data representation, this method provides a new data source for the representation of an outfit.

(5). Improvement of Performance for Body-shape-Aware Recommendation.

This study contributes to the field of body-shape-aware fashion recommendation by introducing a new dataset comprising 20,000 annotated body samples covering five common body shapes. Each body sample includes a 3D body model, anthropometric data, and a frontal view image. This dataset is valuable for body-shape-aware recommendation and related tasks such as virtual try-on and clothed human generation. Additionally, the study proposes the *Body-shape-Aware* network, a novel approach that leverages visual features extracted from body images to enhance the body shape embedding.

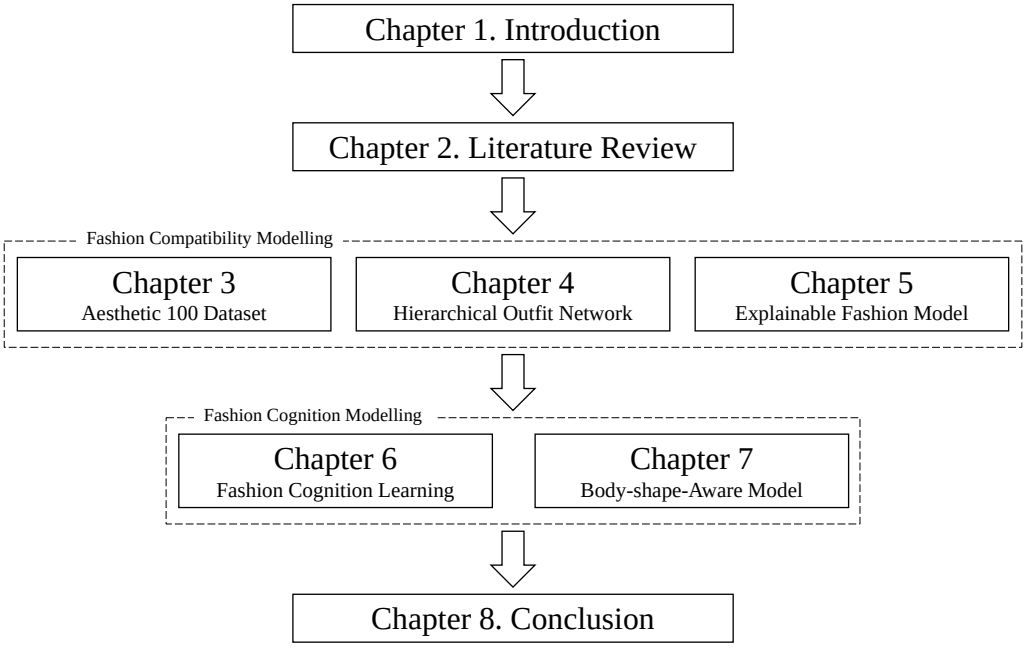


Figure 1.5 The structure of this thesis.

1.5 Organization of the Thesis

The outline of this thesis is illustrated in Figure 1.5. Chapter 1 presents the research background, objectives, and significance of this thesis, providing an introduction to the overall topic and setting the context for the research.

Chapter 2 provides a detailed literature review on fashion recommendation systems. It covers the state-of-the-art methods in the field, available datasets used for evaluation, and the evaluation metrics commonly employed. This chapter serves as a comprehensive overview of the existing research in fashion recommendation.

Chapters 3, 4, and 5 focus on developing a practical fashion compatibility model. Chapter 3 introduces a more comprehensive evaluation protocol named Aesthetic 100, which aims to evaluate the aesthetic quality of fashion

recommendations. Chapter 4 delves into the development of a hierarchical outfit network that leverages the relationships between fashion attributes, items, and outfits to enhance the performance of the fashion compatibility model. Chapter 5 introduces an approach to provide convincing explanations for the fashion compatibility model's prediction.

Chapters 6 and 7 propose two models to solve the task of fashion cognition modeling. Chapter 6 introduces this task and a new dataset. Chapter 7 proposes a body-shape-aware embedding approach to recommend outfits tailored to customers with different body shapes.

Chapter 9 concludes the thesis by summarizing the essential findings and limitations. Additionally, an overview of future works is provided.

Chapter 2

Literature Review

The fashion recommendation system has two essential requirements: **compatibility** and **personalization**. Compatibility ensures that fashion outfits demonstrate aesthetic harmony and coherence among their clothing items. Personalization emphasizes the need to tailor the recommendation process to individual preferences and characteristics, ensuring that the recommended outfits align with each user's unique body and style. In the past years, a multitude of methods, datasets, and evaluation metrics have been proposed to address these two tasks. This chapter presents a detailed review of prior research efforts in fashion recommendation systems.

2.1 Fashion Compatibility Modeling

Fashion compatibility modeling (FCM) is a task that differs from simple fashion item retrieval because FCM considers the aesthetic principles that govern the compatibility of different clothing items in an outfit. This section reviews FCM from two perspectives: the state-of-the-art methods and datasets used

for training and evaluation.

2.1.1 State-of-the-art Methods

At the outset of fashion recommendation research, the main objective is to recommend single fashion items. Iwata *et al.* [56] introduced a fashion recommender system, utilizing a *Probabilistic Topic Model* to acquire knowledge about fashion combinations and suggest an appropriate item according to the query item. Liu *et al.* [76] later developed a practical system called *Magic Closet*, providing users with occasion-oriented item recommendations.

However, the scope of fashion recommendations has expanded beyond individual item suggestions. There is a growing interest in recommending complete outfits, driven by the need for a more comprehensive range of applications. Based on the encoding methods employed on outfits, these methods of FCM can be categorized into three distinct categories: **item-wise modeling**, **graph modeling**, and **try-on modeling**.

Item-wise modeling approaches have been extensively explored in the literature. Viet *et al.* [137] trained *Siamese Networks* using co-occurrence information to predict item-wise compatibility. To address the conflicting similarities between items within a unified space, the authors introduced *Conditional Similarity Networks* in [136]. Vasileva *et al.* [134] proposed a type-specific embedding space that respects item types to learn the similarity and compatibility between items. Wang *et al.* [143] learned fashion compatibility by comparing features from different layers of a convolutional neural network. Lu *et al.* [84] introduced type-dependent hashing modules that consider both item-item and user-item relationships to generate binary codes. Tan *et al.* [128] proposed *Similarity Condition Embedding Network*, which avoids the reliance

on explicit labels during testing by learning representations of similarity under different conditions. Lin *et al.* [71] introduced a *Category-based Subspace Attention Networks*, which is capable of capturing multiple dimensions of similarities. Additionally, the *outfit ranking loss* is commonly employed to improve the representation of item relationships within an outfit.

Graph modeling methods have also been explored for outfit modeling in the literature. These methods leverage the inherent relationships and dependencies among fashion items within an outfit to capture compatibility. Han *et al.* [41] utilized *bidirectional Long Short-term Memory Networks* to estimate the compatibility score based on the probability of the following item given the previously observed items. Li *et al.* [68] introduced the *fashion outfit scoring model*, which employed non-parametric element-wise reduction as a pooling function to handle outfits with varying numbers of items. Cui *et al.* [18] employed a *Node-wise Graph Neural Network* to model outfit compatibility using information from multiple modalities. Cucurull *et al.* [17] applied a *Graph Neural Network* to learn compatibility conditioned on context.

Try-on modeling provides an alternative approach to encoding outfits. It leverages the try-on images to represent outfits, which aligns with the natural inclination of humans to evaluate an outfit visually. Dong *et al.* [25] proposed a *Try-on-guided Compatibility Network* that jointly learns the interaction between individual items and try-on images. They employed a *Multi-modal Try-on Template Network* to generate the try-on appearance automatically. Zheng *et al.* [166] developed a *Collocation and Try-on Network* to solve the fashion compatibility task. They proposed a distillation learning scheme to combine all clothing items for the try-on appearance. By incorporating try-on images, these try-on modeling methods enhance the modeling of outfit compatibility and provide a more visually realistic representation of outfits.

Alongside fashion images, incorporating **fashion attributes** enhances the performance of fashion compatibility models. Zou *et al.* [169] incorporated multiple fashion attributes, such as color, design, and print, to learn fashion compatibility. Feng *et al.* [29] introduced a *Partitioned Embedding Network* that considers color, shape, and texture attributes. Yang *et al.* [152] presented an *Attribute-based Interpretable Compatibility* framework, which predicts compatibility scores based on the attributes of fashion items and pairwise matching relationships. Lu *et al.* [83] employed a stacked self-attention-based method for modeling high-order interactions between items. Ak *et al.* [1] proposed *Attribute Activation Maps* to learn attribute representations. Yang *et al.* [150] utilized the attention mechanism, adaptively evaluating the compatibility score by inferring attribute-level matching signals. In summary, previous research has predominantly focused on encoding outfits at either the item level [128, 134] or attribute level [29, 150], failing to capture the holistic and interconnected nature of fashion data, limiting their evaluation performance.

Incorporating fashion attributes has also prompted research efforts toward **providing explanations** for evaluation results, which is crucial for practical applications. Zou *et al.* [169] proposed to penalize the gradient of the judgment loss to learn the evaluation reasons for the outfit only containing the top and bottom. Plummer *et al.* [104] aimed to explain compatibility between the pair of items by introducing the salient attributes. This approach combined a saliency map highlighting important regions in the image with the most relevant attribute that explains the match. Li *et al.* [67] introduced an attribute-aware recommendation system that can provide fine-grained explanations based on the extracted fashion attribute representations. De *et al.* [20] proposed a memory network exploiting the shape and color attribute features extracted from item images. However, these methods are constrained to pro-

Table 2.1 Summary of the benchmark datasets for fashion compatibility modeling task.

Dataset name	# of fashion types	# of outfits	Source
What-to-Wear [76]	2	24,417	Shopping website
Hu *et al.* [54]	3	28,417	Polyvore
Journey Outfit [162]	-	3,392	Travel review website
Maryland [41]	8	21,889	Polyvore
Li *et al.* [68]	4	195,262	Polyvore
FashionVC [121]	2	20,726	Polyvore
Type-aware [134]	19	68,306	Polyvore
Polyvore-T [143]	5	19,835	Polyvore
Shop the Look [60]	10	38,111	Pinterest
Evaluation3 [169]	2	18,108	Polyvore

cess outfits with a fixed number of items, such as only considering the top and bottom garments. Moreover, a notable limitation is the misalignment between the explanations provided by these approaches and the fashion domain knowledge. For example, the Salient Attribute for Network Explanation [104] derives explanations from image properties rather than learning from a dataset that encompasses fashion-specific knowledge.

2.1.2 Benchmark Datasets

Table 2.1 provides an overview of benchmark datasets utilized for the fashion compatibility modeling task. The What-to-Wear [76] dataset is specifically constructed for an occasion-oriented recommendation, encompassing annotations for full-body, upper-body, and lower-body clothing. Hu *et al.* [54] collected a dataset containing item images and text information, such as categories, names, and descriptions. Journey Outfit dataset [162] contains travel images of several travel destinations designed for the task of Trip Outfits Ad-

visor. The Maryland [41] dataset comprises items accompanied by images and text descriptions. Li *et al.* [68] collected image, title, and category information for each item, along with the number of likes for fashion outfits. The FashionVC [121] dataset is annotated with category, title, and description information. The Type-aware [134] dataset provides annotations for item types, titles, and descriptions. Polyvore-T [143] is a type-label dataset that includes five categories: top, bottom, shoes, bag, and accessory. The Shop the Look dataset, introduced by Kang *et al.* [60], contains scene-product pairs accompanied by bounding boxes for the products and provides information about the product categories. Evaluation3 [169] grades outfits into three levels (poor, normal, and good) and annotates reasons for judgments, such as color, print, and design. These benchmark datasets serve as valuable resources for evaluating and advancing fashion compatibility modeling techniques.

2.2 Personalized Fashion Recommendation

This task involves recommendations based on the user's **preferences** and **physical attributes**.

2.2.1 State-of-the-art Methods

User preferences can be gleaned from various sources, including social media and e-commerce platforms, where data such as outfit likes, user comments, and purchase records provide valuable insights. Packer *et al.* [95] utilized the visual preferences of individual users. They employed interpretable image representations obtained through a unique feature learning process to understand users' previous feedback and their affinity towards specific visual attributes and styles. Wen *et al.* [144] constructed a knowledge graph to recommend

items based on the user's context, such as occasion, weather, and requirements. Chen *et al.* [8] connected user preferences regarding individual items and outfits using Transformer architecture. Kim *et al.* [63] leveraged user-posted outfits in the Polyvore-U dataset [84] to build a knowledge distillation framework for outfit recommendation. They aimed to distill the knowledge from a large dataset into a smaller one for the efficient recommendation. Liu *et al.* [78] incorporated visual information into the rating prediction function and introduced a topic model to categorize words in item reviews into two groups. One group consists of non-visual words explaining the coherent features of the item. In contrast, the other group consists of visual words associated with their visual appearances during different periods. Lu *et al.* [84] considered fashion styles for personalized recommendation by encoding users into binary codes. Li *et al.* [66] proposed a graph network to hierarchically learn relationships among items, outfits, and users. They refined the representation of users through their historical outfits. Sagar *et al.* [111] developed an interpretable compatibility model that captures item-item and item-user interactions. They encoded users into an embedding space using their IDs for personalized recommendations. However, the existing research primarily focused on examining the connection between user preferences and fashion outfits, relying solely on user comments and purchase records to train personalized fashion recommendation systems. They overlooked the significance of incorporating the user's physical characteristics into the modeling process.

As a result, there has been relatively little exploration of modeling relationships between outfits and human physical attributes. Hidayati *et al.* [47, 48] learned the compatibility of clothing styles and body shapes through a set of celebrities' images. Hsiao *et al.* [52] developed a visual body-aware embedding framework to capture the correlation between clothing items and a wide range

Table 2.2 Summary of the benchmark datasets for personalized recommendation.

Dataset name	# of fashion types	# of outfits	Source
Styles and Substitutes [87]	-	773,465	Amazon
He *et al.* [45]	3	28,417	Polyvore
Style4BodyShape [48]	-	347,948	Image search
IQON3000 [122]	6	308,747	IQON
Polyvore-630 [84]	3	127,326	Polyvore
BodyFashion [24]	-	75,695	Amazon
Yu *et al.* [156]	-	208,814	Polyvore
Zheng *et al.* [165]	6	>178,000	Lookbook

of body shapes. Motivated by the observation that different customers pay attention to different aspects of clothing items, Zheng *et al.* [165] introduced a user-specific item-to-set metric trained on personal social media data. This approach aims to personalize the recommendation based on the individual's preference. However, these studies only investigated body shapes as a physical attribute, neglecting other attributes such as hairstyle and skin color, which are valuable for a comprehensive understanding of outfit-person compatibility.

2.2.2 Benchmark Datasets

The benchmark datasets used in personalized recommendation tasks are summarized in Table 2.2. The Styles and Substitutes dataset [87] includes product images, categories, and co-purchase information, offering valuable insights of user preferences and purchase behavior. He *et al.* [45] constructed a comprehensive dataset encompassing users' review histories, purchase histories, and thumbs-up information, providing an understanding of user preferences and interactions. The Style4BodyShape dataset [48] combines data on female celebrities, their body measurements, and photos, enabling exploration of the relationship between body shape and fashion styles. Song *et al.* [122] intro-

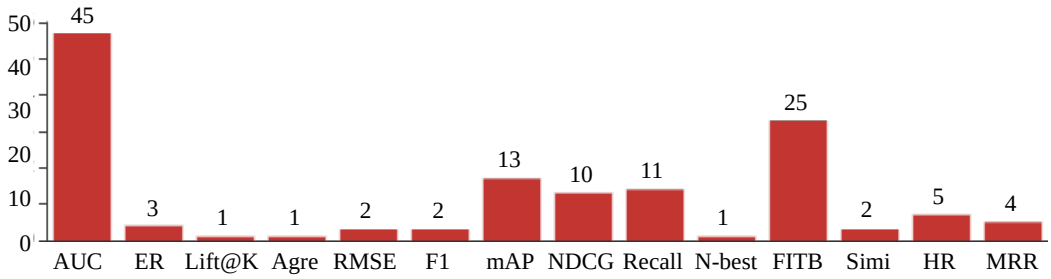


Figure 2.1 Statistics of evaluation indicators adopted in previous fashion recommendation papers. Word abbreviations are employed in the figure for a clear illustration. *Agre* represents *Agreeable* [57]; *N-best* represents *N-best Accuracy* [56]; *Simi* represents *Similarity* [56].

duced the IQON3000 dataset, which contains 308,747 outfits created by 3,568 users. An image, category, attributes, and description accompany each fashion item in the dataset. Additional information, such as price and number of likes, has also been collected for each outfit. Polyvore-630 [84] offers a diverse range of outfit styles and preferences, containing 127,326 outfits created by hundreds of users, accompanied by user information. The BodyFashion dataset [24] includes 11,784 users and their latest historical purchase records, providing a total of 116,532 user-item records that contain information on purchase size and rating. Yu *et al.* [156] constructed a dataset of 208,814 outfits comprising only top and bottom, created by 797 users. Zheng *et al.* [165] collected personal profiles from 2,293 users, including age, number of likes, fans, and garment images cropped using a detection model. While these datasets provide valuable insights into user preferences and purchase behavior, they do not explicitly capture the intricate relationships between fashion outfits and individual physical characteristics such as body shape and hairstyle.

2.3 Evaluation Metrics

Figure 2.1 presents statistics of evaluation indicators adopted in previous fashion recommendation papers. The earliest paper was published in 2007 [114],

while the most recent paper was published in 2023 [20]. Among the total 103 papers, 78 papers present quantitative outcomes. There are 14 different evaluation indicators employed in the previous studies, which are: *Area Under Curve* (AUC) [137], *Error Rate* (ER) [87], *Lift@K* [105], *Agreeable* [57], *Root Mean Squared Error* (RMSE) [21], *F1 score* [145], *mean Average Precision* (mAP) [68], *Normalized Discounted Cumulative Gain* (NDCG) [54], *Recall* [38], *N-best accuracy* [56], *Fill-in-the-Blank (FITB) Accuracy* [41], *Similarity* [56], *Hit Ratio* (HR) [10], and *Mean Reciprocal Rank* (MRR) [73]. Among these 14 evaluation metrics, AUC is the most frequently used indicator, accounting for 36% of fashion recommendation tasks. Specifically, given a positive item pair in the testing set denoted as $(h_i, t_{ig}) \in \mathcal{P}_t$, and N negative items $\{t_{in}\}_{n=1}^{N}$ that belongs to complementary fashion types with h_i, the AUC is calculated using the following function:

$$\text{AUC} = \frac{1}{N\,|\mathcal{P}_t|} \sum_i \sum_n \delta(s(h_i, t_{ig}) > s(h_i, t_{in})) \tag{2.1}$$

where $\delta(x)$ denotes the indicator function, which outputs 0 if the input x is false and one otherwise. $s(h_i, t_{ig})$ calculates the score between this item pair. $|\mathcal{P}_t|$ is the number of all positive pairs. However, the AUC metric only reveals the similarity of the aesthetic tastes of models to those defined in the training set. Other indicators, such as NDCG, mAP, and *Recall*, are also popular in recommending tasks. NDCG and mAP are used for ranked retrieval. Additionally, indicators like MRR, *F1 score*, and *Lift@K* reflect the model's ranking performance. Since Lift@K is not well-known, its definition is given as follows:

$$\text{Lift@K} = \frac{AP@K\,(model)}{AP@K\,(random)} \tag{2.2}$$

ER is adopted in [87] to reflect the relationship of "also-bought". *Agreeable* refers to the degree to which solid and patterned queries agree with the recommendation algorithm's output. Another metric named *N-best accuracy* is defined as the rate of correct recommendation of the top (bottom) out of N recommendations. The *Similarity* evaluates the average degree of similarity between the held-out paired clothing and the recommended clothing. However, the above three indicators are rarely used in later research because of the more complicated tasks [56, 57, 87]. Han *et al.* [41] first introduced the term FITB as selecting an item that matches best with a given outfit. FITB is suitable for tasks that require a connection between semantic and visual information. However, the lack of previous research on objective evaluation in modeling fashion compatibility [92, 172] has resulted in the inadequacy of existing quantitative indicators to assess the aesthetic capabilities of trained fashion compatibility models effectively. The existing evaluation protocols also fail to provide detailed performance insights at more detailed fashion dimensions, limiting their effectiveness.

2.4 Chapter Summary

This literature review explores the field of fashion recommendation systems, focusing on two aspects: compatibility and personalization.

The state-of-the-art methods of fashion compatibility modeling are reviewed, including recommending individual fashion items and complete outfits. Item-wise modeling approaches, such as Siamese networks and conditional similarity networks, have been extensively explored to capture item-wise compatibility. Graph modeling methods leverage the relationships and dependencies among fashion items within an outfit. The try-on modeling methods provide

a visually realistic representation of outfits. Incorporating fashion attributes and using explainability techniques have also been investigated to enhance the interpretability and performance of fashion compatibility models.

Various methods are discussed regarding personalized fashion recommendations. Some methods leverage user-specific data from social media and e-commerce platforms, such as outfit likes, comments, and purchase records. Additionally, some studies have explored the correlation between clothing items and body shapes to provide personalized recommendations.

This chapter also summarizes benchmark datasets for evaluating fashion compatibility modeling and personalized recommendation systems. These datasets provide valuable resources for training and evaluating different algorithms and models in the field.

Lastly, section 2.3 provides an overview of the most commonly used evaluation indicators in previous studies, which focus on either the recommending or the retrieval performance of the fashion recommendation problem.

Chapter 3

How Good Is Aesthetic Ability of a Fashion Model?

3.1 Introduction

Fashion compatibility modeling aims to measure the fashion compatibility of an outfit consisting of variable fashion items [10, 17, 49, 51, 83, 153, 160, 169]. Online retailers leverage the aesthetic capabilities of these fashion compatibility models for cross-selling purposes, encompassing activities such as outfit generation, evaluation, and recommendation. Therefore, a reliable and objective evaluation metric for assessing these models is critical for practical fashion-oriented application. Present methodologies for assessing predominantly prioritize performance evaluation metrics such as mAP [68], MRR [73], Recall [38]. The commonly adopted metric for assessing compatibility classification accuracy is the Area Under the Curve (AUC) [137]. The FITB (Fill-In-The-Blank) accuracy is also introduced in [41] as the evaluation metric. However, none of the existing metrics specifically emphasize capturing the

aesthetic ability of the model, as pointed out in [92, 172].

The term "aesthetic ability" refers to the model's capacity to comprehend the compatibility of fashion items and perceive their aesthetic appeal. In the domain of fashion, the aesthetic system can be categorized into two primary standards, which are **Bottom-up** and **Top-down** [27, 28, 43]. The bottom-up standard implies that the fashion trend emerges from the general population and gradually impacts the mainstream. It represents a collective consensus that occurs when a significant number of individuals agree on a particular fashion trend. On the other hand, the top-down standard originates from professional expertise and will be broadly embraced by the public due to its inherent essence. This standard adheres faithfully to the principles of beauty as established through domain knowledge. It can be viewed as luxury fashion, where predefined styles exert significant influence, leading the public to adopt them. Therefore, three considerations should be taken into account when designing an evaluation protocol for fashion aesthetics:

(1). **Completeness**: The evaluation protocol should establish an objective consensus as the reference basis for quantitative assessment. It should incorporate a systematic standard to ensure a comprehensive evaluation.

(2). **Practicality**. The evaluation protocol should adopt a feasible approach to carry out the assessment. It should consider the practical constraints and limitations to ensure the protocol can be effectively implemented.

(3). **Reliability**. The evaluation protocol should consist of professional and reliable criteria. The evaluation content should be trustworthy and based on authoritative sources to ensure accurate and credible results.

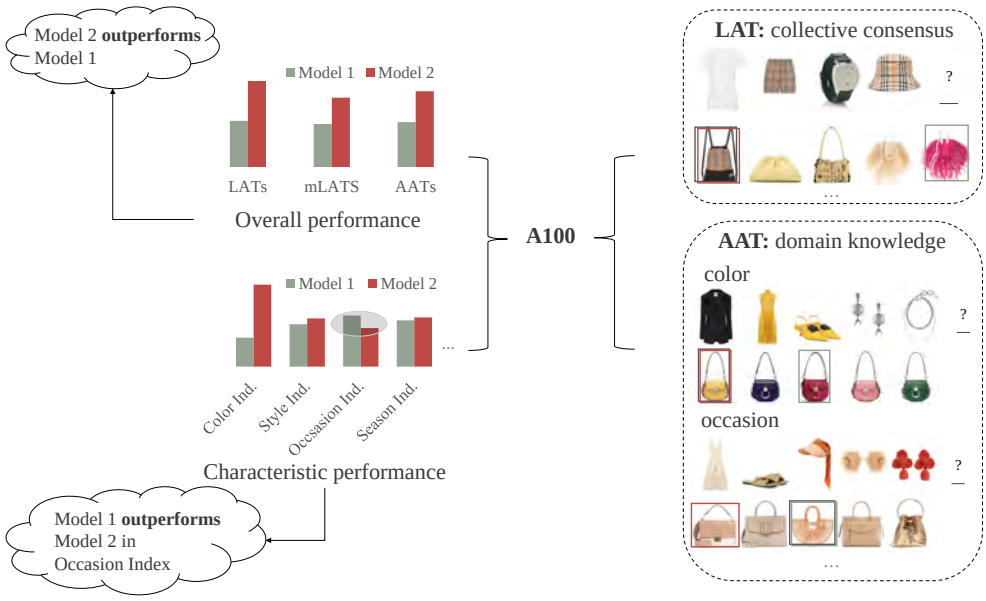


Figure 3.1 The Aesthetic 100 (A100) is introduced to evaluate fashion compatibility models. The LAT and AAT tests of A100 correspond to two aesthetic standards in the fashion domain. The A100 can assess the characteristic performance that may not necessarily align with the overall performance.

To achieve this goal, as depicted in Figure 3.1, a more comprehensive evaluation protocol called **A100** (Aesthetic 100) is proposed, which differs from previous assessments that solely focused on overall performance. Specifically, A100 consists of two tests which are Liberalism Aesthetic Test (**LAT**) and Academicism Aesthetic Test (**AAT**), corresponding to the bottom-up and top-down standards, respectively. Both these two tests are in the form of multiple-choice questions. Images of LAT are gathered from the widely used fashion datasets [41, 121, 134]. The questions are automatically generated and then manually reviewed by fashion experts. Each question has only one correct answer and a collective consensus for the answer has already been achieved in a small group. To collect the ground truth, a website is developed to dis-

tribute the LAT as a questionnaire among the fashion community. Due to the variability in participant responses, the LAT contains two kinds of scores: 1). LAT score (LATs), a challenging score that aligns with the consensus of the majority. The option with the most votes in each question is assigned one point, while all other options are assigned zero points. 2). mean LATs (mLATs), a soft score that considers the minority. Each option's score is the probability of it being chosen.

Pertaining to AAT, the construction process heavily relies on the expertise and involvement of the fashion community due to its high level of professional requirements. After a thorough investigation and in-depth discussions, it has been concluded that six dimensions of the model's aesthetic ability are examined, which are *Color*, *Style*, *Occasion*, *Season*, *Material*, and *Balance*. The AAT's questions and choices are then carefully formed to adhere to these six dimensions. Each question is restricted to examining the single aesthetic dimension of the model, allowing A100 to showcase the model's **characteristic performance** at a fine-grained level. The accuracy of LAT is represented by the AAT score (AATs). The accuracy on each dimension is defined as a specific index, such as *Color* index.

The remainder of this chapter is structured as follows: Chapter 3.2 provides an overview of the related work on fashion compatibility evaluation. Subsequently, Chapter 3.3 describes the construction steps of LAT and AAT in detail. In Chapter 3.4, the experimental results of A100 are presented to validate its effectiveness. Finally, Chapter 3.5 concludes this chapter by summarizing the essential findings.

3.2 Related Work

For practicality purposes, the A100 is formulated as the Fill-in-the-Blank (FITB). To provide the necessary background on FITB, this section presents a summary of the FITB tests that have been utilized in previous studies. Han *et al.* [41] published the first fashion dataset focusing on fashion compatibility in 2017, named Maryland dataset, consisting of 3,076 outfits gathered from the Polyvore website. For each question in the Maryland FITB test, there is an outfit with three incompatible items that are randomly selected from the dataset. Vasileva *et al.* [134] improved the Maryland dataset by respecting item types and proposed the type-aware FITB test set containing 10,000 questions. The incompatible options are randomly selected while they share the same fashion types as the correct option. Lu *et al.* [84] introduced an outfit dataset called Polyvore-U, including user information, and it also employs the FITB test for evaluation. A large dataset named iFashion created by fashion experts from Taobao contains 1.01 million outfits [8]. Ten percent of the outfits are split as the test set to obtain the FITB test. Same as for the Maryland dataset, three incorrect choices are randomly selected from other outfits. In summary, there are two limitations of the previous FITB test set:

(1). **Completeness**. These FITB tests lack uniformity in their aesthetic standard. Various online customers contribute outfits in these datasets, so none of these issues has likely reached a consensus, such as whether an outfit is "compatible" and whether each question's answer is "correct". Additionally, two aesthetic standards (Bottom-up and Top-down) are not covered by these FITB tests.

(2). **Reliability**. These FITB questions are low quality because their choices

are created by randomly sampling several images. Therefore, the correctness of the randomly selected items cannot be guaranteed.

More details can be found in Appendix A.

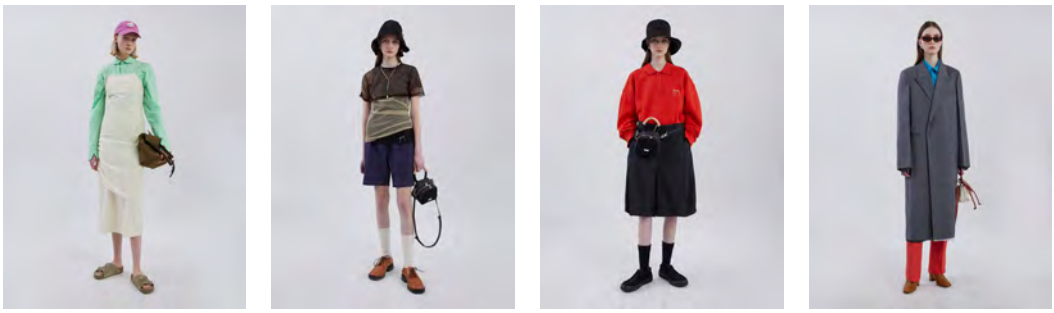
3.3 A100 Evaluation Protocol

The outfit generation principle and construction details of A100 are introduced in this section

3.3.1 Outfit Generation Principle

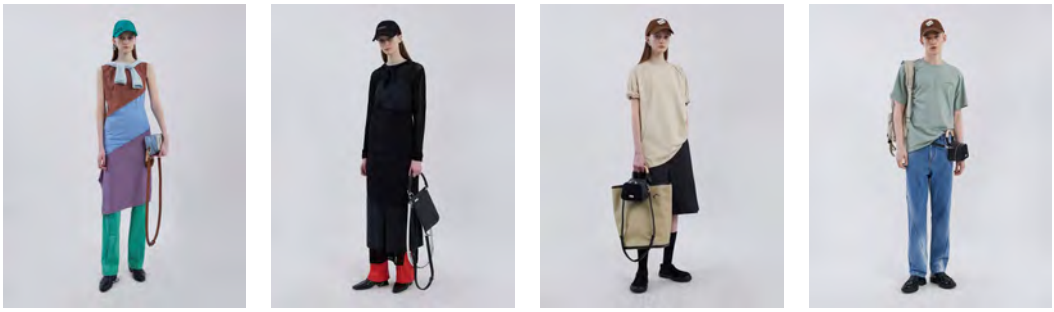
A valid outfit should consist of clothing covering the whole body and shoes. (Four examples are shown in Figure 3.2 (a)). Bags and accessories belong to the complementary items (shown in Figure 3.2 (b)). Clothing is a general term that includes *top*, *pants*, *skirt*, *dress*, *jumpsuit*, and *outerwear*. Accessories include *ring*, *earring*, *necklace*, *bracelet*, *hat*, *watch*, *eyewear*, *legwear*, *glove*, *neckwear*, *brooch*, and *hair wear*. Clothing and shoes are indispensable for a valid outfit. It is ensured that each outfit consists of one and only one pair of shoes, while the categories of pants, skirt, jumpsuit, and dress are mutually exclusive. Some mix-and-match situations are neglected due to lacking the universality, such as multi-layer (Figure 3.2 (c)) and special ways of mix-and-match (Figure 3.2 (d)). Only one item from each clothing subcategory and accessory subcategory can exist in an outfit. Bags and accessories are optional for a complete outfit. An outfit typically consists of a single bag, disregarding specific situations such as Figure 3.2 (e). Thus, in practice, the number of items in an outfit usually does not exceed eight [69], and the number of clothing items in the FITB question is limited to between one and seven. Furthermore,

(a) top and bottom (or one piece) with a pair of shoes



(b) basic outfit in (a) with additional fashion items

(c) utilizing multiple garments to create layers



(d) dress and pants (e) multiple bags

Figure 3.2 Examples of valid outfits.

the number of options is changed from four to five because it is observed that using five options to create these questions achieves a compromise between the workload and test complexity.

3.3.2 Liberalism Aesthetic Test (LAT)

Based on the insights above, the LAT is constructed as following steps:

Step 1: Collecting source images. Three popular fashion datasets are cleaned including FashionVC Dataset [121], Type-aware Dataset [134], and Maryland Dataset [41]. (Each image in A100 depicts one single apparel with a plain background to be consistent with other outfit datasets.) (Note that every image in the A100 includes a single item on a white background consistent with other popular datasets. It takes into account the potential domain shifting across different datasets.) These datasets are collected from websites, and there are many decorative images in these datasets. The decorative images, images with multiple items, and cluttered backgrounds are removed. Then, these images are categorized into 20 categories as summarized in Table 3.1. After rigorous categorization, a pool of clothing items is obtained with the clothing categories. In addition, because these datasets were released a few years ago, 8,972 item images are newly gathered from Mytheresa[1] to keep up with the trends. Combining all the datasets, a vast pool of 366,176 fashion images is obtained.

Step 2: Generating valid outfits. The Outfit Generation Principle (OGP) is proposed to generate outfits automatically to minimize the impact of individual preference on the data. In accordance with the description of a valid outfit defined in Section 3.3.1, clothing and a pair of shoes are first selected,

[1]https://www.mytheresa.com/

Table 3.1 Number of items for 20 clothing categories in the cleaned Maryland dataset [41], cleaned Type-aware dataset [134], cleaned FashionVC dataset [121], and newly collected Mytheresa dataset.

Category	Cleaned-M	Cleaned-T	Cleaned-F	Mytheresa
Tops	19,397	26,528	9,537	1,405
Pants	8,957	12,653	4,703	833
Skirts	5,307	8,592	4,102	527
Jumpsuits	296	820	5	154
Dresses	7,480	12,649	2,607	1,922
Outerwear	10,169	14,172	2,368	961
Bags	21,268	34,882	6	719
Shoes	20,135	38,961	0	687
Rings	3,227	6,265	0	212
Earrings	5,508	12,450	0	123
Necklaces	4,664	7,781	0	352
Bracelets	5,189	7,522	0	207
Hats	2,913	5,550	0	196
Watches	2,290	3,505	0	28
Eyewear	6,685	8,990	1	156
Legwear	202	507	0	155
Gloves	386	723	0	86
Neckwear	1,189	2,778	0	189
Brooch	995	280	0	8
Hair wear	962	1,048	0	52

and then the items belonging to complementary categories are selected. Since unusual scenarios are excluded, such as a skirt with pants or a dress with pants, the jumpsuit and dress can be jointly referred to as one-pieces, and the pants and skirt can be collectively referred to as lower-body. The detailed algorithm of the Outfit Generation Principle is shown in Algorithm 1.

Step 3: Verifying valid outfits. While these generated outfits are valid, they may not necessarily be regarded as compatible outfits. So it is essential to manually review and evaluate them with the assistance of professional stylists.

Algorithm 1 Generating valid outfits

Data: Clothing $\mathcal{C}$, Tops $\mathcal{C}_t$, Bottom $\mathcal{C}_{sp}$, Outerwear $\mathcal{C}_o$, One-piece $\mathcal{C}_{dj}$, Accessories $\mathcal{A}$, Bag $\mathcal{B}$, Shoes $\mathcal{S}$

Result: n complete outfits $\mathcal{O}$

$i = 1$ **while** $i \leq n$ **do**
  α = random.randint(2, 8) ▷ Length of the outfit $\mathcal{O}_i \leftarrow \mathcal{O}_i \cup \mathcal{S}_{rand(1)}$ ▷ The subscript $rand(1)$ denotes random selecting 1 elements from the set
  $\alpha = \alpha - 1$ **switch** α **do**
    case 1 **do**
      $\mathcal{O}_i \leftarrow \mathcal{O}_i \cup (\mathcal{C}_{dj})_{rand(1)}$ ▷ One-piece
    end
    case 2 **do**
      $c = (\mathcal{C} - \mathcal{C}_o)_{rand(1)}$ $\mathcal{O}_i \leftarrow \mathcal{O}_i \cup c$ ▷ Clothing excepting Outerwear **if** $c \in \mathcal{C}_t$ **then**
        $\mathcal{O}_i \leftarrow \mathcal{O}_i \cup (\mathcal{C}_{sp} \cup \mathcal{C}_{dj})_{rand(1)}$
      else if $c \in \mathcal{C}_{sp}$ **then**
        $\mathcal{O}_i \leftarrow (\mathcal{C}_t)_{rand(1)}$
      else
        $\mathcal{O}_i \leftarrow (\mathcal{C}_t \cup \mathcal{C}_o \cup \mathcal{B} \cup \mathcal{A})_{rand(1)}$
      end
    end
    otherwise do
      $c = \mathcal{C}_{rand(1)}$ ▷ Select one clothing $\mathcal{O}_i \leftarrow \mathcal{O}_i \cup c$ **if** $c \in \mathcal{C}_t$ **then**
        $\mathcal{O}_i \leftarrow \mathcal{O}_i \cup (\mathcal{C}_{sp} \cup \mathcal{C}_{dj})_{rand(1)}$ $\mathcal{O}_i \leftarrow \mathcal{O}_i \cup (\mathcal{C}_o \cup \mathcal{B} \cup \mathcal{A})_{randOP(\alpha-2)}$ ▷ The subscript $randOP(n)$ denotes random selecting one element from each category (excepting the $\mathcal{A}$) in the set
    end
    else if $\mathcal{C}_{rand(1)} \in \mathcal{C}_{sp}$ **then**
      $\mathcal{O}_i \leftarrow \mathcal{O}_i \cup (\mathcal{C}_t)_{rand(1)}$ $\mathcal{O}_i \leftarrow \mathcal{O}_i \cup (\mathcal{C}_o \cup \mathcal{B} \cup \mathcal{A})_{randOP(\alpha-2)}$ **else**
        $\mathcal{O}_i \leftarrow \mathcal{O}_i \cup (\mathcal{C}_t \cup \mathcal{C}_o \cup \mathcal{B} \cup \mathcal{A})_{randOP(\alpha-1)}$
      end
  end
  $i = i + 1$
end

To avoid introducing bias into the data, a coarse-to-exemplary approach is implemented. Specifically, five experts are invited to remove the incompatible outfits from the 50,000 randomly generated outfits (10,000 for each expert). After the verifying process, there are 12,096 outfits remaining. The experts are then tasked with assigning an aesthetic score between 1 and 10 to each outfit. Finally, the top 2,000 outfits are selected based on these aesthetic scores.

Step 4: Creating questions with multiple choices. To ensure objectivity, the initial FITB questions are created using these 2,000 outfits and randomly blank one clothing item for each outfit. Each question's wrong choices are randomly selected from the remaining items sharing the same category as the blanked item. Each question contains only one correct answer to ensure effectiveness. These 2,000 questions are then released to ten team members (including those five experts) to complete the test. The advantage of introducing new members is that it can reduce possible bias caused by the first five experts who have seen the questions in the process of verifying. The questions are ranked based on the answers' consistency, and the top 500 questions are retained. Subsequently, the team members are assigned to answer these questions every three days. The order of choices in each question is manually altered, and this process is repeated three times. After careful evaluation, 100 questions are chosen that exhibit 100% consistency in answers. It is important to note that the selected questions consider the balance of different categories, following the ratio of Top : Bottom : Outerwear : One-piece : Bags : Shoes : Accessories = 1 : 1 : 1 : 2 : 2 : 2 : 1.

Step 5: Collecting responses from the crowd. Lastly, a website containing the LAT test is developed, with responses collected from the fashion community. The LAT score (LATs) was defined as follows:

Figure 3.3 An example question in LAT where the red box indicates the ground truth answer. These numbers refer to the proportion of participants who select the corresponding answer ($COUNT(Anw_{model(n)})/x$) as defined in Equation 3.2.

$$LATs = \sum_{n=1}^{100} \delta_{Anw_{model(n)} Max(Anw_{expert(n)})/100} \tag{3.1}$$

where $Anw_{model(n)}$ is the answer index predicted by a compatibility model for the n-th question and δ_{ij} is the Kronecker delta function. The $Max(Anw_{expert(n)})$ refers to the ground truth answer, which is mostly chosen by the public. The mLATs that reveal a variance in people's fashion aesthetics is also provided. The mLATs is defined as follows:

$$mLATs = \sum_{n=1}^{100} COUNT(Anw_{model(n)})/x \tag{3.2}$$

where the function $COUNT(\cdot)$ indicates how many people choose the corresponding answer, and x is the total number of stylists participating in this survey. For example, as depicted in Figure 3.3, it can be observed that the first candidate holds the most significant proportion (0.709), indicating it as the preferred choice among the majority. The remaining candidates' proportions are relatively minor, suggesting a collective consensus consistent with the previously mentioned **Bottom-up** standard.

3.3.3 Academicism Aesthetic Test (AAT)

The design of the Academicism Aesthetic Test is different from LAT. Close collaboration with nine fashion designers is established to incorporate their professional experience, thus meeting the demanding requirements of designing such a test. The creation process follows the "Top-down" aesthetic standard, and the steps involved are presented as follows.

Step 1: Determining Assessing Dimensions. Extensive research and discussions are conducted to determine the specific aspects of the model's artistic ability that need to be examined. Given the textbook's absence of predefined answers, this task is challenging. Previous studies [26, 128, 134] suggest various factors, such as material, color, and style, that have been identified as potentially influencing outfit compatibility. These aspects serve as valuable references for guiding the examination process. In light of this, a thorough investigation is conducted on the FITB questions in the Type-aware dataset [134]. Each question is analyzed in detail to determine the factors that influenced the correctness or incorrectness of the choices. Figure 3.4 illustrates

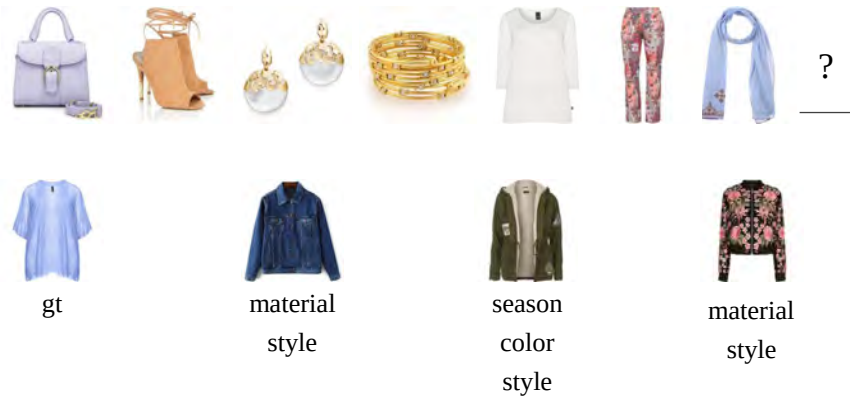


Figure 3.4 A FITB example in the Type-aware test set. It can be observed that more than two factors cause the incorrect answers.

Table 3.2 Specified factors for evaluating the model's ability of fashion aesthetic.

Dimension	Sub-dimension	Question Number
Color	Same Color	1 - 5
	Warm Tone	6 - 10
	Cool Tone	11 - 15
	Contrast Color	16 - 20
Style	Street Wear	21 - 24
	Modern	25 - 28
	Vintage	29 - 32
	Sweet	33 - 36
	Sporty	37 - 40
	Classic	41 - 44
	Gender Neutral	45 - 48
	Mash-up	49 - 52
Occasion	Formal	53 - 55
	Cocktail	56 - 58
	Smart Casual	59 - 61
	Casual	62 - 64
	Holiday	65 - 67
Season	Spring	68 - 70
	Summer	71 - 73
	Autumn	74 - 76
	Winter	77 - 79
Material	Element	81 - 83
	Pattern	84 - 87
	Texture	88 - 91
Balance	Silhouette	92 - 94
	Simple & Complicated	95 - 97
	Proportion	98 - 100

one example question in the Type-aware dataset. It can be observed that multiple factors contribute to the incorrectness of the answers. Specifically, the army-green cotton outerwear is incompatible with the other fashion items due to factors such as season (Winter wear vs. Spring wear), color (incompatibility with the taro-purple bag), and style (casual outerwear vs. elegant style of the other items). After several discussions, the remaining factors are organized and incorporated into a tree structure, resulting in six main dimensions to assess the model's aesthetic ability, where the details are shown in Table 3.2.

Figure 3.5 An example question in AAT where the red box indicate the ground truth answer. The difference between the candidates in the options can only be the dimension designed to examine (color in this case).

Step 2: Creating outfits with styling ideas. A group of nine designers searches for new styling ideas by collecting images from various online websites, such as SSENSE. They create 450 outfits, with 50 outfits contributed by each designer. Subsequently, a voting mechanism is employed to select the top 100 outfits from the entire collection.

Step 3: Designing options accordingly. Based on the sub-dimensions defined in Table 3.2, the examination aspect of each question is determined in advance. For instance, the first 20 questions are assigned to assess the model from the *Color* dimension. Through this strategy, AAT can intuitively demonstrate the compatibility model's specific performance. The performance on the *Color* and *Style* questions are denoted as the color and style indexes, respectively. Note that the proportion of each dimension considers both importance and balance. When creating the incorrect answers, two criteria are adhered to: 1) Each question has only one correct answer; 2) The incorrect answer is solely incorrect due to the predefined factor. This ensures that the evaluation can effectively highlight the model's shortcomings. For instance, as shown in Figure 3.5, the correct option for this example is "the third one". The reason

is that the correct item's color matches the outfit's overall composition. It can be observed that the incorrect answers are identical to the correct answer in all dimensions except for color dimension. The model performance on AAT is defined as AATs (AAT score).

3.4 Analysis

The characteristic of the proposed evaluation protocol is demonstrated through the qualitative and quantitative results. Specifically, two research questions are addressed in this section.

- **Reliability**. Is the evaluation protocol accurate and objective?
- **Explainability**. How does A100 help to explain the aesthetic ability of fashion compatibility models?

3.4.1 Analysis of Reliability

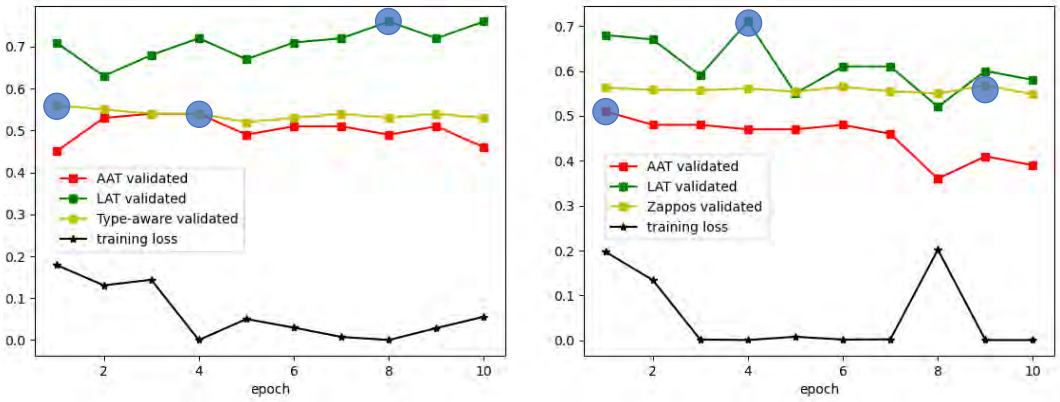
The accuracy of A100 is first examined by comparing the evaluation results of different models on A100. Specifically, four mainstream approaches focusing on modeling fashion compatibility are compared, including Fashion Hashing Network (FHN) [84], Bidirectional Long Short-term Memory Networks (Bi-LSTMs) [41], Type-aware Similarity Embedding (TSE) [134], and Similarity Condition Embedding Network(SCE-Net) [128]. These four models are firstly evaluated on three mainstream FITB tests: the Maryland, Polyvore-630, and Type-aware FITB test. When the models' performances revealed by these tests conflict, a voting mechanism is adopted for judgment. It is worth noting that all models are trained using default parameters and settings because their

Table 3.3 FITB evaluation results of four state-of-the-art methods on the Maryland FITB test [41], Polyvore-630 FITB test [84], Type-aware FITB test [134], LAT test, and AAT test. For TSE [134], its pre-training model is employed directly. As for Bi-LSTMs [41], FHN [84], and SCE-Net [128], they are all retrained according to the open-source codes.

Methods	Training dataset	Maryland	Polyvore-630	Type-aware	LATs	mLATs	AATs
Bi-LSTMs	Maryland	53.50%	41.68%	37.46%	36%	30.82%	35%
FHN	Polyvore-630	46.20%	**53.13**%	45.80%	54%	41.62%	40%
SCE-Net	UT-Zappos50k	51.30%	42.92%	51.53%	72%	54.63%	42%
TSE	Type-aware	**54.97**%	47.07%	**57.69**%	**73**%	**56.17**%	**59**%

training conditions and input data are different. For the sake of fairness, these models are regarded as off-the-shelf models.

The quantitative results are reported in Table 3.3. It is observed that the ranking of four methods, from best to worst, is as follows: TSE, SCE-Net, FHN, and Bi-LSTMs. The performance results assessed on the A100 metric align with this order, demonstrating its accuracy in assessing the performance of various compatibility models. Furthermore, the second column of Table 3.3 indicates that Bi-LSTMs, FHN, SCE-Net, and TSE are trained on distinct datasets, which introduces the risk of overfitting their respective training data. For instance, When evaluated on the Maryland test set, the Bi-LSTMs method demonstrates competitive results, with TSE being the only method that outperforms it. However, the Bi-LSTMs method achieves the lowest performance on the Type-aware and Polyvore-630 test set. This poses a challenge for model generalization and transfer learning, as evaluating the models on the same dataset they are trained on may lead to biased or false-positive performance comparisons. To address this, additional experiments were carried out on the POG dataset [8], which is a dataset distinct from Polyvore. The outcomes of these experiments align with previous findings,



(a) training on Type-aware dataset

(b) training on UT-Zappos50k dataset

Figure 3.6 Experiments of finding the optimal model for TSE [134] and SCE [128] based on AAT (red), LAT (green), and the Type-aware validation set (yellow) to verify the objectivity of A100. Blue circles on curves represent optimal stop epochs.

as the results indicate the following rankings: Bi-LSTMs (47.21%) < FHN (54.13%) < SCE-Net (57.27%) < TSE (66.65%). To further verify the reliability of A100, other baselines such as NGNN [18] and CAS-Net were also evaluated. NGNN, trained on the Maryland dataset like Bi-LSTMs, achieved a FITB accuracy of 50.68% compared to Bi-LSTMs' 53.50%, LATs of 33% compared to 36%, mLATs of 29.88% compared to 30.82%, and AATs of 30% compared to 35%. These results show that A100 is an effective standalone protocol for evaluating fashion compatibility models.

To verify the objectivity of A100, a comparative experiment is conducted using a validation set. The TSE model is retrained using the same settings on the identical training dataset. To determine the optimal training epoch, the model's performance is assessed using three distinct validation sets: the original validation set, LAT (from A100), and AAT (from A100). These checkpoints saved at each optimal epoch can be used to compare the model's performance on multiple widely-used testing datasets. Figure 3.6 (a) depicts the

Table 3.4 The performance results of optimal TSE [134] retrained on Type-aware dataset [134], and SCE-Net [128] retrained on UT-Zappos50k dataset [128]. All results reported below are FITB accuracy.

Training dataset	Models	Maryland	Polyvore-630	Type-aware	LATs	mLATs	AATs
Type-aware	AAT validated	56.93%	44.31%	53.69%	69%	52.76%	**64%**
	LAT validated	**57.41%**	**46.42%**	52.05%	**76%**	**57.65%**	58%
	Type-aware validated	56.17%	42.12%	**55.58%**	71%	54.81%	57%
UT-Zappos50k	AAT validated	**53.84%**	**43.87%**	**55.46%**	**79%**	**63.19%**	**52%**
	LAT validated	52.11%	42.64%	51.63%	75%	59.42%	49%
	Zappos validated	51.21%	41.44%	49.57%	71%	54.17%	42%

validation and training losses at each epoch. The blue circles on the red, yellow, and green curves indicate the optimal epoch determined by AAT (epoch 8), Type-aware (epoch 1), and LAT (epoch 4), respectively.

The quantitative results of the three models are presented in Table 3.4. Among them, the Type-aware validated model excels in the Type-aware test, but consistently underperforms in other indicators. This finding emphasizes the effectiveness of A100 in identifying models with better generalization. Figure 3.6 (b) depicts a similar experiment with SCE-Net trained on the UT-Zappos50k training set, which yields consistent patterns as shown in Table 3.4. Notably, the optimal model obtained through AAT validation consistently demonstrates superior performance across all test sets

Table 3.5 AAT results of FHN [84], Bi-LSTMs [41], TSE [134], and SCE-Net [128]. Each index score is obtained by calculating the ratio of correct answers to the total number of questions in the index. Take *Balance* index for example, the score is computed by dividing the number of correct answers in the *Balance* group by 9.

Indexes	FHN	Bi-LSTMs	TSE	SCE-Net
Color (20%)	0.50	0.30	**0.85**	0.75
Style (32%)	0.22	0.34	**0.50**	0.28
Occasion (15%)	**0.60**	0.40	0.53	0.33
Season (12%)	0.42	0.50	**0.58**	0.33
Material (12%)	0.50	0.42	0.50	0.50
Balance (9%)	0.33	0.11	**0.56**	0.33
AATs	0.40	0.35	**0.59**	0.42

3.4.2 Analysis of Explainability

I. Quantitative analysis.

To validate the A100's explainability, Table 3.5 presents the detailed performances of FHN, Bi-LSTMs, TSE, and SCE-Net. The AAT design methodology, introduced in Section 3.3.3, allows A100 to examine the model's aesthetic ability on six fine-grained dimensions. Each dimension contains a different number of questions ensuring a balanced representation of sub-dimensions. Some interesting insights can be observed in Table 3.5:

(1). TSE outperforms other methods primarily due to its exceptional performance on the *Color* index, where TSE achieves **0.85**, whereas FHN and Bi-LSTMs only achieve 0.5 and 0.3, respectively.

(2). FHN demonstrates a better understanding of *Occasion* than TSE and SCE-Net. This can be attributed to its training data of user information being more closely related to the *Occasion* dimension, leading to better performance in this aspect.

Table 3.6 Performances comparison of FHN [84], Bi-LSTMs [41], TSE [134], and SCE-Net [128] on the *Season* dimension.

	FHN	**Bi-LSTMs**	**TSE**	**SCE-Net**
Spring	1	2	**3**	1
Summer	**2**	1	1	1
Autumn	1	**2**	1	1
Winter	1	1	**2**	1
Total	5	6	**7**	4

(3). FHN has the lowest score in the *Style* index among the three models, with a score of only **0.22**. This can be attributed to the inclusion of diverse users' information, which broadens the impact of different personal preferences and complicates the model's comprehension of *Style*.

(4). Bi-LSTMs obtain a score of **0.11** in the *Balance* dimension, which primarily focuses on assessing the shape of clothing items, such as *Proportion* and *Silhouette*. This outcome suggests that Bi-LSTMs exhibit lower sensitivity toward the item's shape.

Through these analyses, it can be concluded that the A100's characteristic performance offers a more comprehensive perspective for evaluating models compared with conventional evaluation metrics.

Table 3.6 compares performances of these four methods on *Season* dimension, which consists of four sub-dimensions: *Spring*, *Summer*, *Autumn*, and *Winter*. TSE demonstrates the highest overall performance among these models, while SCE-Net appears to underperform on the *Season* dimension. More specifically, the Bi-LSTMs method exhibits good performance in the *Autumn* group, FHN achieves the highest scores in the *Summer* group, and TSE showcases relatively strong ability in the *Spring* and *Winter* groups.

Figure 3.7 Examples of AAT assessing performances of models on *Color* index. (a). Question examining *Same Color*. (b). Question examining *Contrast Color*.

II. Qualitative analysis.

Figure 3.7 provides a further illustration of the qualitative findings regarding the *Color* dimension. This dimension contains *Cool Tone*, *Warm Tone*, *Contrast Color*, and *Same Color*. Upon analyzing the qualitative results, there are several observations:

(1). Bi-LSTMs method exhibits a limited capacity for matching colors. As evidenced in Figure 3.7 (a), the choice of green boots selected by Bi-LSTMs is deemed unreasonable given the colors of the items mentioned in the question. In contrast, TSE consistently demonstrates strong performance within the color group. Among the 20 questions, all the incorrect answers selected pertain to the *Contrast Color* subgroup.

(2). SCE-Net achieves a *Color* index score of 0.75, with the four incorrect questions also falling into the *Contrast Color* subgroup. An example is illustrated in Figure 3.7 (b). The army green boots are the most suitable choice as they form an interesting color composition with the red bag and dress.

(3). FHN performs relatively poorly on this dimension, getting 0.5 in the *Color*

index. It earns 4 points for *Warm Tone* questions while receiving two points for each of the other three sub-dimensions.

Figure 3.8 Example question in LAT. The numerical values below each option represent the selection ratio among individuals.

The results of these four models obtained on LAT aligns with the insights obtained previously. This is evident in Figure 3.8, where it can be observed that the Bi-LSTMs method consistently exhibits a bold taste in color matching, while TSE maintains a good performance in the *Color* dimension. Furthermore, the agreement ratios of participants in selecting each option further validate the observations provided by the AAT. For first question, the Bi-LSTMs method selects the answer with only 4% agreement among participants, whereas TSE chooses the same answer with over 75% agreement. Similarly, in the second question, options selected by Bi-LSTMs and FHN have lower agreement percentages of 4% and 10%, respectively, while TSE selects the answer with over 72% agreement. These observations emphasize the characteristic performance of fashion compatibility models across dimensions, not limited to *Color*.

By analyzing the model's performance across various dimensions, one can

identify the specific aspects where the model requires improvement. For instance, A100 highlights the relatively weak performance of existing models in the *Balance* dimension. To enhance the model, a potential approach could involve sampling more data related explicitly to the balance aspect.

3.5 Chapter Summary

In this chapter, a new evaluation protocol named A100 is introduced to examine the aesthetic ability of the fashion compatibility model. A100 incorporates fine-grained dimensions to examine specific areas where the models may be lacking. By incorporating these evaluations into performance analysis, valuable insights can be gained for improving the models. The comprehensive analysis conducted shows its effectiveness. In the next chapter, a proposed fashion compatibility model is compared against 14 baselines utilizing the A100 evaluation protocol. The new aesthetic perception indicators have the potential to contribute to the development of modern fashion intelligence systems and inspire practical applications in the field of real fashion AI.

Chapter 4

Hierarchical Outfit Network for Fashion Compatibility Learning

4.1 Introduction

The continuous advancement of artificial intelligence (AI) has dramatically accelerated the fashion industry, particularly in the context of online consumption, which has maintained its dominance and continues to grow [4]. In this online landscape, cross-selling plays a crucial role in boosting the click-through rate (CTR) and sales revenue of online retailers. However, one of the main challenges faced by these retailers is intelligently generating qualified fashion compositions for their customers. To address this challenge, there has been a growing body of literature focused on fashion compatibility learning, which aims to learn the compatibility of multiple clothing items in an outfit [19, 83, 91, 160, 170].

Most of the approaches jointly utilize multi-modal information to enhance model performance. Han *et al.* [41] introduces visual semantic embedding

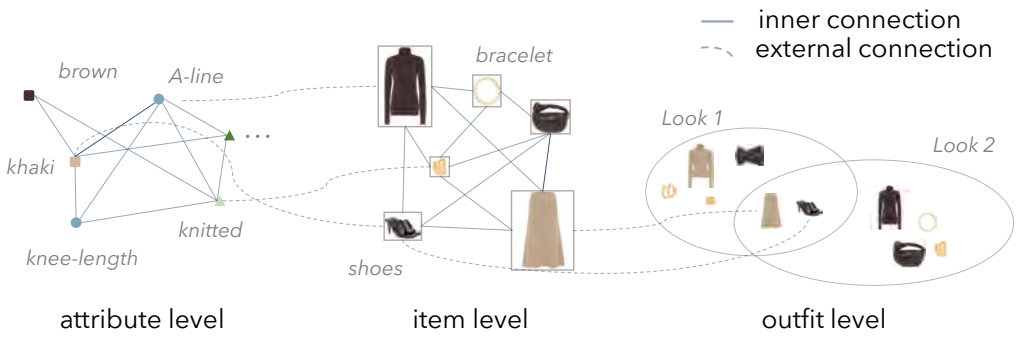


Figure 4.1 Relations of fashion outfits among attribute level, item level, and outfit level

to learn relationships between fashion images and corresponding text descriptions. The type-aware method proposed by [134] suggests incorporating fashion-type information during the process of embedding item features. Lin *et al.* [71] enhance the model performance by introducing the outfit ranking loss, which operates on a whole outfit. However, these approaches overlook the fashion attributes which are imperceptibly utilized by human beings in fashion compatibility evaluation, including color, silhouette, and sleeve length. In other words, fashion attributes play an essential role in modeling fashion compatibility. Some studies [29, 150, 152, 169] argue that incorporating fashion attributes can improve both the performance and explainability of fashion models. Feng *et al.* [29] define the color, texture, and shape as main factors and propose a partitioned embedding network to learn their embeddings. Yang *et al.* [152] proposes an interpretable compatibility method based on fashion attributes such as color, pattern, and shoe type. However, one limitation of these approaches is that they overlook the hierarchical structure among fashion data. As illustrated in Figure 4.1, multi-relations exist among outfits, items, and attributes. Yang *et al.* [152] connected the attribute and outfit level using the informative attribute crosses method. However, they ignored the item level.

Yang *et al.* [150] proposed an attribute-wise explainable model. Although it adopts the *Attribute Activation Map* to learn the attribute-wise representations, the overall compatibility evaluation is computed on the item level, which omits the outfit level.

To fully utilize the hierarchical structure of fashion data, the Hierarchical Outfit Network, termed as **HON**, is proposed in this chapter. The HON can process an outfit containing variable-length items. Specifically, the attention mechanism [135] is leveraged to model hidden relationships among the attribute, item, and outfit levels. Each feature vector is passed to the next level network in the form of aggregation. The HON contains three sub-networks, from bottom to top, namely, Attributes Level Network (ALN), Item Level Network (ILN), and Outfit Level Network (OLN). The attribute level considers the fashion attributes from attribute dimensions, such as color, material, and print. The ALN is designed to generate implicit attribute features from images in a self-supervised manner. These implicit features are then concatenated into explicit attributes at the item level serving as the input of ILN. The explicit fashion attributes refer to features that can be obtained without annotations, such as the primary color. The item level considers the characteristics of the item dimension, such as category, season, and brand. At this level, ILN learns the interactions among attribute features through the attention mechanism and outputs the item embeddings. The outfit level considers the characteristics of outfit dimension, such as style, feeling, and trend. The outfit embedding is obtained at this level after calculating the interactions among all item embeddings. Finally, the outfit embedding is used to calculate the compatibility score of the given outfit.

The remainder of this chapter is structured as follows: Chapter 4.2 provides an overview of the attention mechanism applied to modeling fashion.

Subsequently, Chapter 4.3 describes the structure of the proposed HON in detail. In Chapter 4.4, the effectiveness of HON is validated through extensive experiments. Chapter 4.5 presents qualitative results of utilizing HON for practical applications. Finally, Chapter 4.6 concludes this chapter.

4.2 Related Work

The attention mechanism, the core block of the proposed HON, has been widely recognized for its effectiveness in various tasks. In the context of natural language processing, the attention mechanism was initially introduced in the encoder-decoder framework to address the challenge of compressing variable-length sentences into fixed-length vectors for text classification tasks [127]. One notable advancement in attention mechanism came with the introduction of the *Transformer* architecture [135], which revolutionized machine translation tasks. The *Transformer* utilized a self-attention mechanism to capture global dependencies in the input sequence, allowing the model to attend to relevant information at different positions.

In the fashion domain, Lu *et al.* [83] employed a stacked self-attention mechanism to model interactions among fashion items. This approach effectively captured the dependencies and relationships among different elements of an outfit. Yang *et al.* [150] utilized the attentive interaction mechanism to incorporate attribute-level matching signals into the overall compatibility evaluation dynamically.

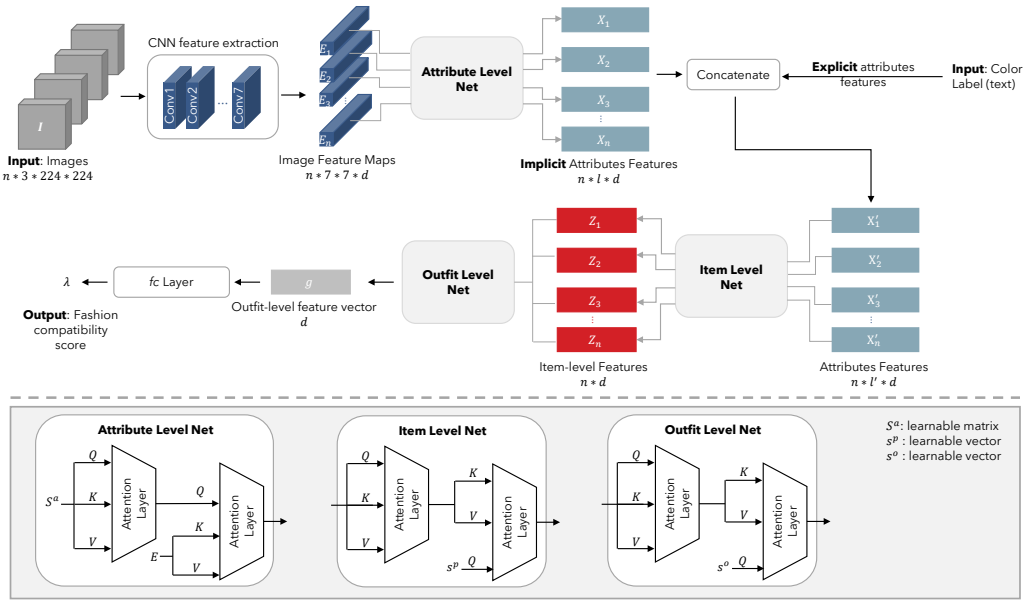


Figure 4.2 The overview of the proposed Hierarchical Outfit Network, consisting of the Attribute Level Network, Item Level Network, and Outfit Level Network. HON takes images and other metadata of an outfit as input and outputs the desired outfit-level embedding, which is used for scoring the outfit's fashion compatibility.

4.3 Hierarchical Outfit Network

The overview framework of the proposed Hierarchical Outfit Network (HON) is illustrated in Figure 4.2. It contains three primary submodules, namely Attribute Level Network (ALN), Item Level Network(ILN), and Outfit Level Network(OLN). The ALN embeds the feature maps extracted from the item images into the attribute-level space. The ILN embeds the attribute features into the item-level space. The attribute features are a combination of the output of ALN, referred to as **implicit** attribute features, and the **explicit** attribute features. The OLN aggregates all item features and outputs the desired outfit embedding. This section first presents the formulation of the fashion compatibility problem. Then, detailed descriptions of the three sub-

modules are introduced, followed by the objective function.

4.3.1 Problem Formulation

Given a set of items $\mathcal{M} = \{p_i\}_i^{N_p}$ of N_p individual items and an outfit collection $\mathcal{T} = \{O_j\}_{j=1}^{m}$ of m outfits, each outfit $O = \{p_i\}_i^n$ in collection $\mathcal{T}$ is defined as a subset of $\mathcal{M}$ containing n items, where $p_i \in \mathcal{M}$. Each item $p_i \in \mathcal{M}$ has its corresponding image I_i and other metadata such as primary color data, category label, and textual data, which varies depending on the datasets. The task of modeling fashion compatibility aims to obtain the compatibility score for a given outfit O via encoding it into an outfit feature vector $\mathbf{g}$. The proposed HON encodes an outfit as follows:

$$\mathbf{g} = \text{HON}(O|\mathbf{\Theta}_{\text{HON}}) \tag{4.1}$$

where $\mathbf{\Theta}_{\text{HON}}$ is the parameters of model to be learned.

4.3.2 Multi-head Attention

A multi-head attention mechanism [135] is employed in the following three networks to obtain the features in the corresponding level. Given a query vector set $\mathbf{Q} \in \mathbb{R}^{n \times d_k}$, a key vector set $\mathbf{K} \in \mathbb{R}^{n \times d_k}$, and a value vector set $\mathbf{V} \in \mathbb{R}^{n \times d_v}$, the attention mechanism outputs the weighted sum of values as follows,

$$\text{Attn}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Softmax}(\frac{\mathbf{Q}\mathbf{K}^{\text{T}}}{\sqrt{d_k}})\mathbf{V} \tag{4.2}$$

Multi-head attention mechanism projects the vector sets $\mathbf{Q}$, $\mathbf{K}$, and $\mathbf{V}$ into h subspaces through different weights. These projected vector sets in subspaces are computed via the attention mechanism in parallel mode. The

output from each subspace will be concatenated and then projected, yielding final results.

$$\begin{aligned} \text{MultiHead}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) &= \text{Concat}(\mathbf{H}_1, \ldots, \mathbf{H}_h)\mathbf{W}^{\mathbf{O}} \\ \text{where } \mathbf{H}_i &= \text{Attn}(\mathbf{Q}\mathbf{W}_i^Q, \mathbf{K}\mathbf{W}_i^K, \mathbf{V}\mathbf{W}_i^V) \end{aligned} \tag{4.3}$$

where $\mathbf{W}_i^Q \in \mathbb{R}^{d \times d_k}$, $\mathbf{W}_i^K \in \mathbb{R}^{d \times d_k}$, $\mathbf{V}_i^Q \in \mathbb{R}^{d \times d_v}$, and $\mathbf{W}^{\mathbf{O}} \in \mathbb{R}^{hd_v \times d}$ are projection matrices.

In HON, dimensions of all vectors are the same, *i.e.* $d_k = d_k = d_v = d$. For the sake of simplicity, the subscripts of these dimensionality notions are omitted. The Attention Layer serves as the fundamental module for each level network and is defined as follows:

$$\text{AttnLayer}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{LayerNorm}(\mathbf{Q} + \text{MultiHead}(\mathbf{Q}, \mathbf{K}, \mathbf{V})) \tag{4.4}$$

4.3.3 Attribute Level Network

Attribute level network is designed to embed feature maps extracted from *convolutional neural network* (CNN) into attribute-level space. Given an image data I which belongs to the item $p \in \mathcal{M}$, the image feature maps are obtained by flattening the output of the final convolutional layer of the CNN:

$$\mathbf{E} = \text{Flatten}(\text{CNN}(\text{I} \mid \mathbf{\Theta}_{\text{CNN}})) \in \mathbb{R}^{\tau \times d} \tag{4.5}$$

where $\tau = 49$ is the flattened dimension. A learnable embedding vector set $\mathbf{S}^a \in \mathbb{R}^{l \times d}$ is used to learn the implicit attribute features for the given item since they are not explicitly learned from any fashion attribute annotations. Here l is a hyper-parameter indicating the number of implicit attributes, and

the superscript a indicates $\mathbf{S}^a$ belongs to the attribute level. The structure of the attribute level network is defined as follows,

$$\begin{aligned} \mathbf{F}^a &= \text{AttnLayer}(\mathbf{S}^a, \mathbf{S}^a, \mathbf{S}^a) \\ \mathbf{X} &= \text{AttnLayer}(\mathbf{F}^a, \mathbf{E}, \mathbf{E}) \end{aligned} \tag{4.6}$$

where $\mathbf{F}^a \in \mathbb{R}^{l \times d}$. $\mathbf{X} \in \mathbb{R}^{l \times d}$ is the output of ALN representing that there are l implicit attribute features with dimension d learned from the given image I.

A fashion item encompasses image data and other metadata, including main color data, category labels, and textual data. So, the complementary information can be separately encoded into attribute features, referred to as explicit attribute features. Concatenating the implicit attribute features learned from ALN and explicit attribute features, the complete attribute features $\mathbf{X}' = \text{Concat}(\mathbf{X}_{\text{imp}}, \mathbf{X}_{\text{exp}}) \in \mathbb{R}^{l' \times d}$ is obtained, where l' is the new length of attributes. In ALN, a fashion item is encoded into l' attributes that serve as the input of the item level network.

4.3.4 Item Level Network

Item level network aggregates complete attribute features $\mathbf{X}'$ into item feature $\mathbf{z}$. A learnable embedding vector $\mathbf{s}^p \in \mathcal{R}^d$ is leveraged to represent the desired item-level feature, where the superscript p indicates that it belongs to the item level. The item level network can be expressed as follows:

$$\begin{aligned} \mathbf{F}^p &= \text{AttnLayer}(\mathbf{X}', \mathbf{X}', \mathbf{X}') \\ \mathbf{z} &= \text{AttnLayer}(\mathbf{s}^p, \mathbf{F}^p, \mathbf{F}^p) \end{aligned} \tag{4.7}$$

where $\mathbf{F}^p \in \mathbb{R}^{l' \times d}$ is the self-attentive output of attribute features, and $\mathbf{z} \in \mathbb{R}^d$ is the learned item feature. By applying ILN, the item feature is obtained after fully considering the inter-relationships between explicit and implicit attribute features through the attention mechanism.

4.3.5 Outfit Level Network

It is worth noting that ALN and ILN are applied to a single item, while an outfit may contain variable-length items. To this end, for the outfit level network, all item features are aggregated into one vector at the outfit level. Given an outfit $O = \{p_i\}_i^n$ with n items, the concatenated item features $\mathbf{Z} = \text{Concat}(\mathbf{z}_1, \ldots, \mathbf{z}_n) \in \mathbb{R}^{n \times d}$ is computed through the ALN and ILN, where $\mathbf{z}_i$ is the output of ILN for item p_i. A learnable vector $\mathbf{s}^o \in \mathbb{R}^d$ is applied for aggregation. The outfit level network can be expressed as follows:

$$\begin{aligned} \mathbf{F}^o &= \text{AttnLayer}(\mathbf{Z}, \mathbf{Z}, \mathbf{Z}) \\ \mathbf{g} &= \text{AttnLayer}(\mathbf{s}^o, \mathbf{F}^o, \mathbf{F}^o) \end{aligned} \tag{4.8}$$

where $\mathbf{F}^o \in \mathbb{R}^{n \times d}$ is the self-attentive output considering the inter-relationship between all items and $\mathbf{g} \in \mathbb{R}^d$ is outfit feature vector in Equation 4.1. Due to the architecture of OLN, HON is capable of handling outfits with various items.

4.3.6 Objective Function

To obtain the fashion compatibility score of the given outfit O, a fully connected layer is added to project the outfit feature vector $\mathbf{g}$ into a scalar $\lambda = \text{FC}(\mathbf{g} \mid \mathbf{\Theta}_{\text{FC}})$. A margin ranking loss is applied to learn the parame-

ters of the network, which is defined as follows:

$$\mathcal{L}(O^+, O^-) = \max(0, (\lambda^+ - \lambda^-) + \alpha) \tag{4.9}$$

where λ^+ and λ^- are compatibility scores of outfits O^+ and O^-, respectively. O^+ represents the positive outfit, while O^- represents the negative outfit, implying that λ^+ should be greater than λ^-. The hyper-parameter α is some margin value.

In practice, every outfit $O \in \mathcal{T}$ is considered positive because they are manually created. On the other hand, the negative outfits are generated according to the following steps based on the positive outfit:

(1). Selecting one item p^+ from the original positive outfit O^+;

(2). Randomly picking one item $p^- \in \mathcal{M}$ that shares the same category with p^+;

(3). Replacing p^+ with p^- from O^+ to generate the negative outfit O^-.

The overall cost function is defined as follows:

$$J(\mathbf{\Theta}) = \sum_{O^+ \in \mathcal{T}} \sum_{p \in \mathcal{O}^+} \mathcal{L}(O^+, O^-) + \frac{\lambda}{2} \|\mathbf{\Theta}\|_2^2 \tag{4.10}$$

where $\mathbf{\Theta}$ contains $\mathbf{\Theta}_{\text{HON}}$, $\mathbf{\Theta}_{\text{CNN}}$,and $\mathbf{\Theta}_{\text{FC}}$. λ is the L2 regularization hyper-parameter.

4.4 Experiments

To validate the effectiveness of the proposed HON, experiments are conducted on two mainstream datasets and the A100 evaluation protocol described in

Chapter 3. There are three research questions to be addressed in this section:

- RQ1: Does the proposed HON outperform the state-of-the-art models?
- RQ2: Does the proposed HON leverage the advantages of the claimed multi-layer relations and the implicit attribute features?
- RQ3: Why are the multi-layer relations helpful to enhance model performance?

4.4.1 Experimental Settings

Parameter settings. HON is trained on the non-disjoint split of Type-aware dataset [134] with a margin of 0.3, dropout of 0.1, and batch size of 60 on NVIDIA A100 GPU. *ResNet-18* is used as the image feature extractor. Both the height and width of the images are cropped to 224. The embedding size of features d is 512, and the number of multi-head h is 4. Adam is chosen as the optimizer with the learning rate, beta, and weight decay as 5×10^{-5}, 0.9, and 10^{-5}, respectively. The learning rate in the training process follows an exponentially decreasing schedule, along with adopting an early stop training strategy. The main colors are extracted from item images by using FOCO system [171] as the explicit feature in the item level network. It facilitates the reduction of dependency on input data and the improvement of practicality.

Evaluation datasets. The newly introduced A100 and two mainstream test sets are employed for evaluation: the Maryland test set [41] and the Type-aware test set [134]. The Maryland dataset contains 21,899 outfits collected from the Polyvore website. There are 17,316 outfits allocated for training, 1,497 allocated for validation, and 3,076 allocated for testing. For each outfit, three incorrect choices are randomly selected from all available products. As

for the Type-aware dataset, it contains two dataset variants which are disjoint and non-disjoint sets. Unlike the strategy used to create the FITB test in [41], it considers clothing categories when sampling wrong options. The non-disjoint set of the Type-aware dataset is adopted for training and testing in this work.

4.4.2 Compared Methods

The HON is compared with the 14 state-of-the-art methods. Methods denoted with the superscript * are trained on their original dataset. In contrast, for the methods without the superscript, their pre-trained models are utilized for testing. All methods are tested on three test sets after necessary adaptation.

WNN* [87] learns the fashion compatibility by using a weighted Euclidean distance over image feature $d_{\mathbf{w}} = \|\mathbf{w} \circ (\mathbf{x}_i - \mathbf{x}_j)\|_2^2$, where $\mathbf{w}$ is a vector with the same dimension as $\mathbf{x}$ and $\circ$ is the Hadamard product.

LMT* [87] transforms image features to low-dimensional space. The distance function is defined as $d_{\mathbf{Y}}(\mathbf{x}_i, \mathbf{x}_j) = \|(\mathbf{x}_i - \mathbf{x}_j)\mathbf{Y}\|_2^2$. A shifted *Sigmoid* function is employed to calculate probability over distance by optimizing the log-likelihood on an observed relationship set.

SiameseNet [137] models the compatibility of an outfit as an average value of item-wise feature similarity extracted from Siamese CNN. The distance function is the cosine distance.

Pooling* [68] aggregates features extracted from multiple items of an outfit by an average pooling operation.

Concatenation* [131] concatenates all features of items as the input.

Bi-LSTMs [41] learns the fashion compatibility by training a bidirectional LSTM to maximize the probability of the following fashion item being conditioned on previously seen items.

Self-attention* [142] uses the scaled dot-product attention [135] to learn re-

lationships between items.

TSE [134] calculates the item-wise similarity respecting item types. Item features are projected to the type-specific embedding space, and the distance function is defined as $d_{ij}^{uv} = \left\|(\mathbf{x}_i^u - \mathbf{x}_j^v) \circ \mathbf{w}^{(u,v)}\right\|_2^2$. The fashion compatibility is learned using a modified triplet loss function.

MCN [143] measures the item-wise similarity in a projected latent space using the cosine distance function. The features generated from different layers of the CNN network are considered different comparison aspects, and a 2-layers predictor computes the compatibility score.

FHN [84] splits the compatibility score concerning the user into two parts: a weighted sum of the user's relationship to the fashion items and the compatibility between pairs of fashion items. Due to the absence of user information, only the pairs-wise item relationships are computed.

NGNN [18] models the fashion compatibility by constructing a *Fashion Graph*. The vertices in the graph represent categories, and the edges represent the relationship between categories. The compatibility score is computed using self-attention on the graph-level output by representing an outfit as a subgraph.

SCE-Net* [128] projects image features to masked embeddings via a masking operation on image feature and similarity condition masks. A condition weight branch is obtained by concatenating two item features as an assignment of the similarity condition masks.

CAS-Net* [71] computes the fashion compatibility concerning a reference item within the given outfit. The category-based attention networks are employed to capture multiple dimensions of similarities.

LPAE-Net* [83] utilizes the item aggregation network to encode several items into a compatibility embedding. The self-attention mechanism is used to model interactions among the items.

Table 4.1 Comparing HON with different methods on the Maryland test set and Type-aware test set. Only 21% and 26% of the data in Maryland and Type-aware test sets are suitable for evaluating methods with the subscript †, respectively, because these methods cannot predict outfits with varied items. The experiments of HON are repeated five times, and the values after ± are the mean square error.

	Maryland test [41]		Type-aware test [134]	
Methods	FITB Acc.	AUC	FITB Acc.	AUC
WNN [87]	31.18%	0.52	30.06%	0.51
LMT [87]	33.22%	0.56	32.38%	0.56
SiameseNet [137]	49.00%	0.82	50.78%	0.83
Pooling† [68]	25.08%	0.51	24.85%	0.50
Concate.† [131]	24.16%	0.71	30.42%	0.67
Bi-LSTMs [41]	53.50%	0.85	37.46%	0.63
Self-atten.† [142]	23.25%	0.58	31.06%	0.58
TSE [134]	54.97%	0.85	57.69%	**0.88**
MCN† [143]	44.07%	0.88	42.43%	0.87
FHN [84]	46.20%	0.68	45.80%	0.70
NGNN [18]	50.68%	0.83	32.64%	0.50
SCE-Net [128]	51.30%	0.79	51.53%	0.80
CAS-Net [71]	55.85%	0.85	55.88%	0.84
LPAE-Net [83]	37.65%	0.65	40.27%	0.71
HON	**60.08 ± 0.42** %	**0.88 ± 0.01**	**59.05 ± 0.93** %	0.85 ± 0.03

4.4.3 Quantitative Results (RQ1)

Results on Mainstream datasets. Table 4.1 compares the performance of HON with 14 state-of-the-art methods on two mainstream datasets in terms of FITB accuracy and AUC. From this table, it can be observed that the proposed HON outperforms the other methods on three of the four indicators. Specifically, for the FITB accuracy metric, HON achieves 60.08% on the Maryland test set and 59.05% on the Type-aware test set, which are significant gains of +4.23% and +3.17% FITB accuracy over CAS-Net which omits the attribute information. Additionally, compared with another attribute-free method, *e.g.*, NGNN of 50.68% on FITB accuracy and 0.83 on AUC, the gains by HON are

Table 4.2 Comparing HON with different methods on the LAT and AAT. There are only 40% and 32% of the data in LATs and AATs are suitable for evaluating methods with the subscript †, respectively. The experiments of HON are repeated five times, and the values after ± are the mean square error.

	LAT		AAT
Methods	LATs	mLATs	AATs
WNN [87]	31%	28.84%	29%
LMT [87]	32%	29.70%	28%
SiameseNet [137]	63%	46.72%	32%
Pooling† [68]	27.50%	26.63%	18.75%
Concatenation† [131]	25%	23.41%	6.25%
Bi-LSTMs [41]	36%	30.82%	35%
Self-attention† [142]	32.50%	28.26%	21.88%
TSE [134]	73%	56.17%	**59**%
MCN† [143]	53%	41.44%	34.30%
FHN [84]	54%	41.62%	40%
NGNN [18]	33%	29.88%	30%
SCE-Net [128]	72%	54.63%	42%
CAS-Net [71]	72%	55.94%	44%
LPAE-Net [83]	31%	20.48%	28%
HON (Ours)	**78**%	**58.20**%	57%

also high, at +9.4% and +0.03 on Maryland dataset. These justify the advance of HON, which takes advantage of utilizing attribute information. Compared with the state-of-the-art item-wise methods, *i.e.*, TSE, HON surpasses it with a clear margin: +5.11% on Maryland FITB test, and +1.36% on Type-aware FITB test. This may be attributed to the fact that HON exploits the multi-relations among fashion data, while TSE only uses the fashion image data. It is also noted that TSE is +0.03 better than HON on the AUC metric of the Type-aware dataset. This may indicate that respecting the category helps improve the AUC performance.

Results on Aesthetic 100 test. The HON is also compared with baseline methods on the proposed A100 in Table 4.2 to demonstrate its superior aes-

Table 4.3 Results of TSE and HON evaluated on the AAT. Indexes refer to the performance on specific dimensions.

Detailed Indexes	**TSE [134]**	**HON**
Color Index (20%)	0.85	0.80
Style Index (32%)	0.50	0.53
Occasion Index (15%)	0.53	0.53
Season Index (12%)	0.58	0.50
Material Index (12%)	0.50	0.50
Balance Index (9%)	0.56	0.44
AATs	0.59	0.57

thetic ability. From this table, it can be concluded that HON outperforms all baselines on LATs and mLATs and achieves competitive results on AATs. Specifically, HON obtains 78% LATs and 58.2% mLATs, surpassing the results of SCE-Net by +6% LATs and 3.75% mLATs. It is also +5% LATs and +2.03% mLATs higher than TSE. This result may be due to TSE trying to model fashion compatibility by measuring item-wise similarities. It can be concluded that the HON has a good generalization ability. The characteristic performances of TSE and HON are further investigated by utilizing the explainability of AAT. The detailed results in the fine-grained dimensions are presented in the last two columns of Table 4.3. It can be observed that HON performs better than the TSE in the *Style* Index while scoring lower in the *Color*, *Season*, and *Balance* indexes.

Table 4.4 Experimental results on different structures of network.

Methods	LAT		AAT	Maryland test		Type-aware test	
	LATs	mLATs	AATs	FITB Acc.	AUC	FITB Acc.	AUC
Drop ALN	56%	44.37%	40%	51.82%	0.81	50.36%	0.79
Drop ILN	68%	53.65%	49%	59.43%	0.87	58.00%	0.84
Drop OLN	77%	57.69%	**56**%	59.87%	0.81	58.55%	0.78
Drop ILN & OLN	70%	53.44%	50%	59.27%	0.81	58.43%	0.77
HON	**78**%	**58.2**%	**56**%	**60.08**%	**0.88**	**59.05**%	**0.85**

4.4.4 Ablation Study (RQ2)

Network Structure. The efficacy of the three sub-networks of HON is examined in this subsection. This ablation study is conducted by removing one or two networks, and the results are reported in Table 4.4. The model performance drops dramatically in all metrics when removing the attribute layer network (Drop ALN row). It shows that the implicit attribute features extracted by ALN are critical for modeling fashion compatibility. In addition, replacing ILN by the maximum pooling layer operating on all attribute features causes severe damage to the model performance on the LAT and AAT metrics (Drop ILN row). However, the performance on the other two test sets shows only a slight decrease. Furthermore, if the OLN is removed, it can be observed that there is also a significant decrease in the model's performance in the metric AUC (-7%/-7%) for the Maryland and Type-aware test set (Drop OLN line). In contrast, dropping OLN has little effect on the performance of LAT and AAT. It indicates that the AUC metric is highly dependent on modeling item interactions. Finally, after removing the ILN and OLN (Drop ILN & OLN line), the result shows a performance reduction on almost all metrics. Interestingly, the negative effect applied to the model performance combines the adverse impacts of dropping ILN and OLN independently.

Table 4.5 Experimental results on different numbers of implicit attributes.

# Implicit	LAT		AAT	Maryland test		Type-aware test	
Attributes	LATs	mLATs	AATs	FITB Acc.	AUC	FITB Acc.	AUC
2	66%	51.04%	47%	59.75%	0.87	58.02%	**0.85**
5	**78**%	**58.2**%	**56**%	60.08%	**0.88**	**59.05**%	**0.85**
10	72%	54.92%	53%	58.55%	0.87	58.06%	**0.85**
25	70%	53.78%	**56**%	59.01%	0.86	58.26%	**0.85**
50	68%	51.84%	51%	**60.6**%	0.87	58.52%	**0.85**

Number of Implicit Attributes. The effect of the number of implicit attributes in the attribute level network is examined, and the results are reported in Table 4.5. It can be observed that the overall performance achieves the best when five implicit attributes exist. This result may imply that too few implicit features cannot capture the attribute relationship among the fashion items. At the same time, too many implicit features may overwhelm the added explicit attribute features. Another interesting observation is that the number of implicit attributes has minor effects on the AUC metric.

4.4.5 Empirical analysis on qualitative results (RQ3)

To further elaborate HON for leveraging the multi-layer relations between attributes, items, and outfits, FITB question examples answers by different approaches are visualized in Figure 4.3. The Q1 in Figure 4.3 indicates the importance of attribute-level information. Both choice C and choice D are black tops. However, choice C is incompatible with this outfit because of its type of sleeves. Specifically, the lantern sleeves design is not compatible with wearing inside a blazer. The Q2 in Figure 4.3 indicates the importance of item-level information. Conditioned on the sweater and high-heel sandals in

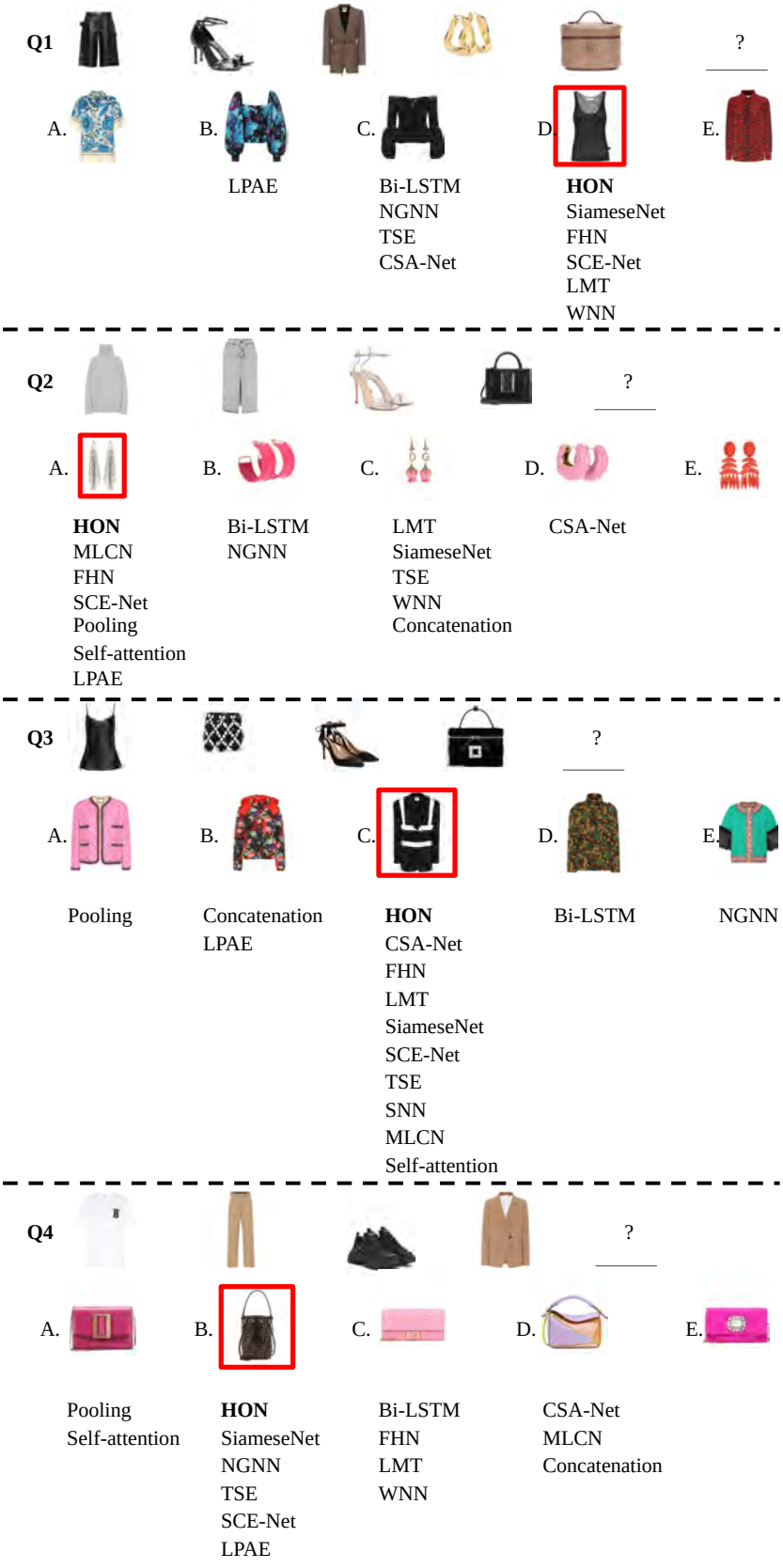


Figure 4.3 Qualitative results of different approaches. The red bounding box indicates the correct answer.

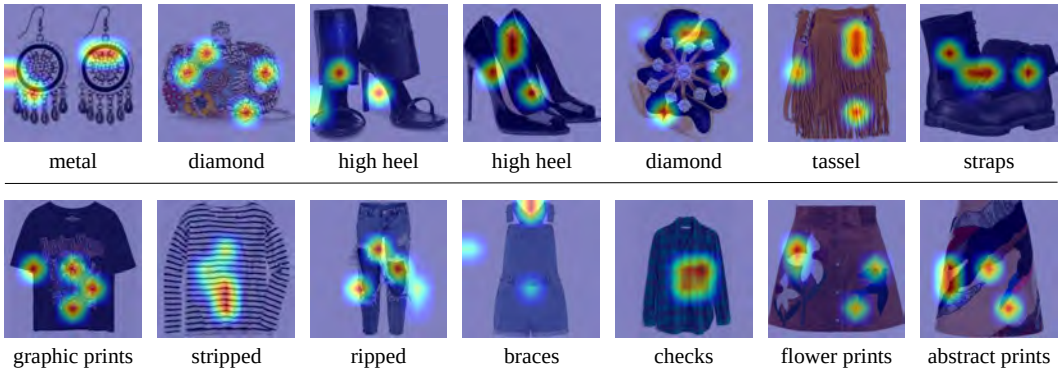


Figure 4.4 Visualization results of implicit attributes (Equation 4.6).

this outfit composition, a white earring is more compatible than other items. The Q3 in Figure 4.3 indicates the importance of outfit-level information. The choices in Q3 are all jackets. However, there is only one correct answer when considering the style of the whole outfit. As shown in Q4, HON consistently performs well when changing the category to bags and can select the correct answer. Based on these qualitative results, we conclude that the HON indeed learns the relations among attributes, items, and outfits, resulting in its improvement compared with previous methods.

Figure 4.4 illustrates the visualization results of the implicit attributes learned via ALN(Equation 4.6). Interestingly, these attention maps may indicate some corresponding regions of fashion attributes, *i.e.*, neckline region of the top, print region of the top, sleeve length of the outerwear, and dress. It demonstrates that our HON utilizes the attribute information hidden in item images to benefit the learning process.

4.5 Application

There are three practical tasks that mainly adopt fashion compatibility models, including: 1). Fashion outfit complimentary item retrieval [71]; 2). Fashion outfit evaluation and revision [169]; and 3). Fashion outfit generation and recommendation [18]. Thus, this section presents these qualitative results with a flowchart of how to utilize our HON for practical applications.

4.5.1 Complementary Item Retrieval

Task: Given an outfit with variable length and a set of items with the same category, the aim is to find the top k most compatible items with the given outfit. Figure 4.5 illustrates the quantitative results obtained by applying the HON model, which involves four sequential steps:

Step 1. Enumerate in the set of items;
Step 2. Combine the chosen item with the outfit to generate a new outfit;
Step 3. Compute the compatibility score using HON and store all scores;
Step 4. Sort all scores in a non-increasing order and find the top k items.

Figure 4.5 Visualized examples of adopting HON on the Fashion outfit complementary item retrieval [71].

Figure 4.6 Visualized examples of adopting HON on the task of Fashion outfit evaluation and revision [169]. The red box indicates the found incompatible item.

4.5.2 Fashion Outfit Evaluation

Task: Given an outfit with variable length, the aim is to find the incompatible item in this outfit and revise it from a set of items with the same category. Figure 4.6 illustrates the quantitative results obtained by applying the HON model, which involves four sequential steps:

Step 1. Compare the score of the given outfit and threshold.

Step 2. Enumerate in the given outfit;

Step 3. Replace the chosen item with a random item from the item set to generate a new outfit;

Step 4. Compute the compatibility score using HON until the score exceeds a threshold.

Step 5. Sort all scores in a non-increasing order and find the top k items.

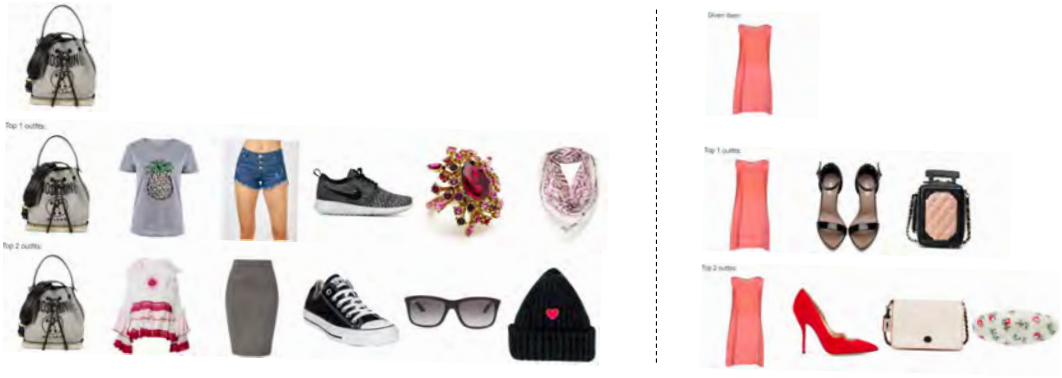


Figure 4.7 Visualized examples of adopting HON on the task of Fashion outfit generation and recommendation [18].

4.5.3 Fashion Outfit Generation

Task: Given an initial item, the aim is to generate k compatible outfits with variable lengths. Figure 4.7 illustrates the quantitative results obtained by applying the HON model, which involves four sequential steps:

Step 1. According to fashion outfit generating logic, randomly choose items from their corresponding category set.

Step 2. Repeat generation process n times;

Step 3. Compute the compatibility scores using HON and store all scores;

Step 4. Sort all scores in a non-increasing order and find the top k outfits.

4.6 Chapter Summary

This chapter focuses on the development of the fashion compatibility model and its application in real online cross-selling scenarios. A novel network called HON is introduced, which exploits the hidden relations among attributes,

items, and outfits to enhance the prediction performance. The experiment demonstrates that HON outperforms 14 baselines and performs state-of-the-art on two widely used FITB tests and the A100 metric. The ablation studies on network structure and implicit attributes prove the effectiveness of the proposed components of HON. Furthermore, strategies for integrating the trained fashion compatibility model into actual products for online cross-selling are proposed. These solutions aim to bridge the gap between research and practical implementation, enabling the effective utilization of the fashion compatibility model in real-world contexts.

Chapter 5

Modeling Fashion Compatibility with Convincing Reasons

5.1 Introduction

The widespread popularity of online shopping platforms has provided people with a vast array of clothing choices and the ability to share their fashion preferences online. However, not everyone possesses the expertise to create a well-coordinated outfit with compatible mix-and-match fashion items. Therefore, there is a significant demand for practical solutions that can offer professional mix-and-match recommendations to users.

Nevertheless, predicting the compatibility of fashion outfits is a complex task in the context of artificial intelligence, as it involves multiple factors such as visual perception, texture, and trend. Previous research has explored various methods for compatibility prediction, with metric learning-based approaches emerging as the mainstream approach [50, 87, 134]. These methods encode fashion items into embeddings and calculate metric distances to deter-

mine compatibility. For instance, the Bi-LSTM network [41] has been utilized to learn compatibility among fashion items by treating the outfit as a sequential input. Other approaches leverage techniques such as *Conditional Random Field* [119] and clothing style modeling [2, 137] to estimate fashion compatibility. However, one limitation of these methods is the lack of explanation for their compatibility predictions, which is crucial for gaining user trust in practical applications.

To address the need for explanations in compatibility predictions, several studies have attempted to provide explanations alongside compatibility predictions. Lin *et al.* [72] offered fashion suggestions and generated abstract comments as explanations simultaneously. Wu *et al.* [146] introduced the *Visual and Textual Jointly Enhanced Interpretable model* to generate interpretable fashion recommendations. Chen *et al.* [12] proposed a *Co-attentive Multi-task Learning model* to generate explainable recommendations. Despite these efforts, most of these approaches rely heavily on comments or reviews from social network users, resulting in training data that lacks insights from fashion experts. Consequently, the explanations provided by these models may be inconclusive and less reliable.

Furthermore, certain studies [143, 169] have limitations in evaluating outfits with multiple items, restricting their flexibility and scalability. Real-world outfits can vary significantly in the number of components they comprise, and fixed item constraints may not capture this diversity effectively. Consequently, the applicability of these methods becomes limited when dealing with outfits that deviate from the fixed item constraint.

To address the above limitations, this chapter proposes a novel fashion compatibility modeling framework to provide convincing explanations aligned with the expert's knowledge. The judgment and reason are jointly trained using the

Table 5.1 Fashion attributes of utilized in providing the prediction reason.

	Top	Bottom	Shoes	Bag
Color	✓	✓	✓	✓
Print	✓	✓	×	×
Material	✓	✓	✓	✓
Design	✓	✓	×	×
Silhouette	✓	✓	×	×
Shoes Upper	×	×	✓	×
Heel Height	×	×	✓	×
Heel Type	×	×	✓	×
Shape of Bag	×	×	×	✓

Bidirectional Long Short-term Memory (Bi-LSTM) networks. Specifically, feature vectors are first extracted from item images through *ResNet* models. The color features are extracted using the FOCO system [171]. Fashion attributes are leveraged to enhance the prediction performance and provide the prediction reason. The detailed fashion attribute information is shown in Table 5.1. Fashion compatibility modeling is learned through a one-layer Bi-LSTM, which can process the outfit containing multiple items. Firstly, all the extracted attribute features are stacked and sent to the model. Subsequently, the item features computed at each step of Bi-LSTM model will be concatenated and input to the inter-factor compatibility network, where the corresponding reason is determined. The inter-factor compatibility network is designed to trace back the gradients of attribute features to obtain the contribution of each element in the decision-making process. Fashion compatibility is categorized into three levels: *Good*, *Normal*, and *Poor*. Through quantitative and qualitative experiments, the developed network demonstrates its ability to accurately predict compatibility levels and provide corresponding reasons.

5.2 Related Work

5.2.1 Explainable Fashion Compatibility Model

Numerous studies have been conducted to evaluate the compatibility of outfits using various approaches [41, 50, 116, 119, 121, 134, 137, 143]. Some of these studies focused on learning visual compatibility through unsupervised methods [51] or measuring scene-product compatibility using CNNs and attention mechanisms [60]. Others explored the synthesis problem, aiming to transform outfits into more fashionable ones [53] or guide the generation process based on compatibility [156]. There were also approaches that explicitly modeled visual compatibility through fashion image inpainting [40] or learned compatibility in a unified space using frameworks like TransNFCM [154] or Relation Networks [93]. A graph neural network called Neural Graph Filtering was also adopted to model fashion collocation [77].

Despite the progress made by these studies, one common limitation is the lack of a convincing explanation for the calculated compatibility. The ability to provide explanations is crucial in fashion compatibility modeling. Explaining the reasoning behind compatibility evaluations helps users understand and trust the system's decisions. It allows users to understand why certain outfits are deemed compatible or incompatible, enhancing their fashion sense and decision-making process.

5.2.2 Long Short-term Memory (LSTM) networks

The LSTM, a variant of an RNN, is introduced to be capable of learning long-term dependencies without suffering the "vanishing gradients". The LSTM has been proven in many applications including speech [36, 37] and video [23, 159].

It has also been applied in the fashion domain. Heinz *et al.* [46] employed LSTM to overcome the cold start problem for the fashion recommendation method. The output layer of Bidirectional LSTM can get information from backward and forward states simultaneously. Han *et al.* [41] learned the compatibility relationships among fashion items using Bi-LSTM. Nakamura *et al.* [94] used Bi-LSTM to extract style information of an outfit.

This chapter adopts a Bi-LSTM network to learn the fashion compatibility between fashion items as an outfit is treated as a sequence. There are two advantages to using Bi-LSTM for modeling outfit compatibility. First, LSTM is not limited to fixed length input, which allows modeling for sequential data of various lengths containing item features, just as text or vision problems. Second, adding or reducing some items in an outfit for Bi-LSTM models is simple and easy to tune parameters end-to-end.

5.3 Approach

Given an outfit, its compatibility judgment is first evaluated, involving three categories: *Good*, *Normal*, or *Poor*. Subsequently, the main reason behind the judgment is predicted by evaluating the contribution of each attribute of the fashion items. It is worth noting that a *Normal* outfit does not have a specific reason associated with it. The set of the reason is denoted as $\mathcal{R} = \{Color, Print, Material, \cdots, Shape\}$, encompassing the different factors that play a role in determining the compatibility of an outfit. Each element in the reason set represents an aggregated attribute, such as combining the colors of the top and bottom garments into the single factor of *Color*. This section introduces three modules of the proposed model: feature extraction architecture, bidirectional LSTM architecture, and gradient penalty architecture.

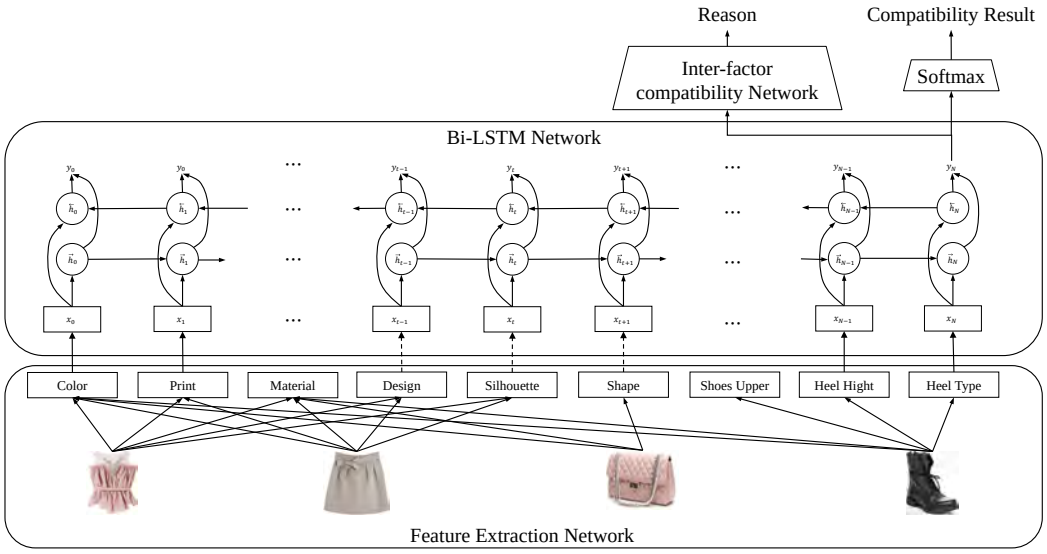


Figure 5.1 The pipeline of fashion compatibility network. The Bi-LSTM networks receive feature maps extracted from CNNs as input, where each feature is considered a contributing factor. The output feature at the last step of the Bi-LSTM is encoded into the compatibility judgment space using a *Softmax* layer. The inter-factor compatibility network assesses the reason for judgment by taking the output features of the Bi-LSTM as inputs. The network employs gradient penalty to facilitate the learning process.

5.3.1 Feature Extraction Architecture

The pipeline of the compatibility network is depicted in Figure 5.1. Given an outfit with multiple items, the feature extraction network is designed to extract various fashion attribute features from the input images. Specifically, *Color* feature is encoded using the color histogram. Five main colors are first extracted using the *Fashion Color System (FOCO)* [171] and then concatenated to form the color attribute features. Apart from the color attribute, a pre-trained *ResNet-18* [44] model is employed to extract other attributes. The final feature map, which is a 512-dimensional vector, is utilized to represent fashion items.

5.3.2 Bidirectional LSTM Architecture

The bidirectional LSTM network, as illustrated in the middle of Figure 5.1, is employed to learn the compatibility of an outfit by leveraging the attribute features extracted in the previous stage. Given an input sequence $\mathbf{x} = (\mathbf{x}_1, \cdots, \mathbf{x}_T)$, a standard recurrent neural network (RNN) computes the hidden vector sequence $\mathbf{h} = (\mathbf{h}_1, \cdots, \mathbf{h}_T)$ and output vector sequence $\mathbf{y} = (\mathbf{y}_1, \cdots, \mathbf{y}_T)$ by iterating the following equations from $t = 1$ to T:

$$\begin{aligned} \mathbf{h}_t &= \mathcal{H}(\mathbf{W}_{ih}\mathbf{x}_t + \mathbf{W}_{hh}\mathbf{h}_{t-1} + \mathbf{b}_h + \tau) \\ \mathbf{y}_t &= \mathbf{W}_{ho}\mathbf{h}_t + \mathbf{b}_o \end{aligned} \tag{5.1}$$

where $\mathbf{W}_{ih}$ is the input-hidden weight matrix and $\mathbf{b}_h$ is the hidden bias vector. $\mathcal{H}$ represents the hidden layer activation function.

The feature sequence of an item is denoted as $\mathbf{F} = (\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N)$, where $\mathbf{x}_t$ is the embedding feature extracted using a CNN model for the t-th attribute. The Bi-LSTM Network performs several calculations to generate the desired outputs. The Bi-LSTM Network iterates through the backward layer from $t = T$ to 1 and the forward layer from $t = 1$ to T. During this process, it computes the *forward* hidden sequence $\overrightarrow{\mathbf{h}}$, the *backward* hidden sequence $\overleftarrow{\mathbf{h}}$, and the output sequence y using the following equations:

$$\begin{aligned} \overrightarrow{\mathbf{h}}_t &= \mathcal{H}(\mathbf{W}_{x\overrightarrow{h}}\mathbf{x}_t + \mathbf{W}_{\overrightarrow{\mathbf{h}}\overrightarrow{h}}\overrightarrow{\mathbf{h}}_{t-1} + \mathbf{b}_{\overrightarrow{h}}) \\ \overleftarrow{\mathbf{h}}_t &= \mathcal{H}(\mathbf{W}_{x\overleftarrow{h}}\mathbf{x}_t + \mathbf{W}_{\overleftarrow{h}\overleftarrow{h}}\overleftarrow{\mathbf{h}}_{t+1} + \mathbf{b}_{\overleftarrow{h}}) \\ \mathbf{y}_t &= \mathbf{W}_{\overrightarrow{h}y}\overrightarrow{\mathbf{h}}_t + \mathbf{W}_{\overleftarrow{h}y}\overleftarrow{\mathbf{h}}_t + \mathbf{b}_y \end{aligned} \tag{5.2}$$

where $\mathbf{W}_{\boldsymbol{\alpha\beta}}$ is the weight matrix between vector $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$. $\mathbf{b}$ represents the

bias term and $\mathcal{H}$ is the Long Short-Term Memory (LSTM) cell. Specifically, $\mathcal{H}$ is implemented by the following composite function:

$$\begin{aligned}
\mathbf{i}_t &= \sigma(\mathbf{W}_{xi}\mathbf{x}_t + \mathbf{W}_{hi}\mathbf{h}_{t-1} + \mathbf{W}_{ci}\mathbf{c}_{t-1} + b_i) \\
\mathbf{f}_t &= \sigma(\mathbf{W}_{xf}\mathbf{x}_t + \mathbf{W}_{hf}\mathbf{h}_{t-1} + \mathbf{W}_{cf}\mathbf{c}_{t-1} + b_f) \\
\mathbf{c}_t &= \mathbf{f}_t\mathbf{c}_{t-1} + \mathbf{i}_t\tanh(\mathbf{W}_{xc}\mathbf{x}_t + \mathbf{W}_{hc}\mathbf{h}_{t-1} + b_c) \\
\mathbf{o}_t &= \sigma(\mathbf{W}_{xo}\mathbf{x}_t + \mathbf{W}_{ho}\mathbf{h}_{t-1} + \mathbf{W}_{co}\mathbf{c}_t + b_o) \\
\mathbf{h}_t &= \mathbf{o}_t\tanh(\mathbf{c}_t)
\end{aligned} \tag{5.3}$$

where σ is the logistic sigmoid function, and $\mathbf{i}$, $\mathbf{f}$, $\mathbf{o}$, and $\mathbf{c}$ are the *input gate*, *forget gate*, *output gate*, and *cell* activation vectors, respectively. $\mathbf{W}$ is the weight matrix and b is the bias term. A *softmax* layer is used to compute a separate output distribution $\Pr(k|t)$ at each step t along the input sequence as follows:

$$\begin{aligned}
\mathbf{y}_t &= \mathbf{W}_{\overrightarrow{h}y}\overrightarrow{\mathbf{h}}_t + \mathbf{W}_{\overleftarrow{h}y}\overleftarrow{\mathbf{h}}_t + \mathbf{b}_y \\
\Pr(k|t) &= \frac{\exp(\mathbf{y}_t[k])}{\sum_{k'=1}^{K}\exp(\mathbf{y}_t[k'])}
\end{aligned} \tag{5.4}$$

where k is the number of the judgments, and $y_t[k]$ is the k-th element of the output vector $\mathbf{y}_t$. The loss for a given outfit F can be calculated as:

$$L_{\text{judgment}} = -\frac{1}{K}\sum_{k=1}^{K}\log\Pr(k|t) \tag{5.5}$$

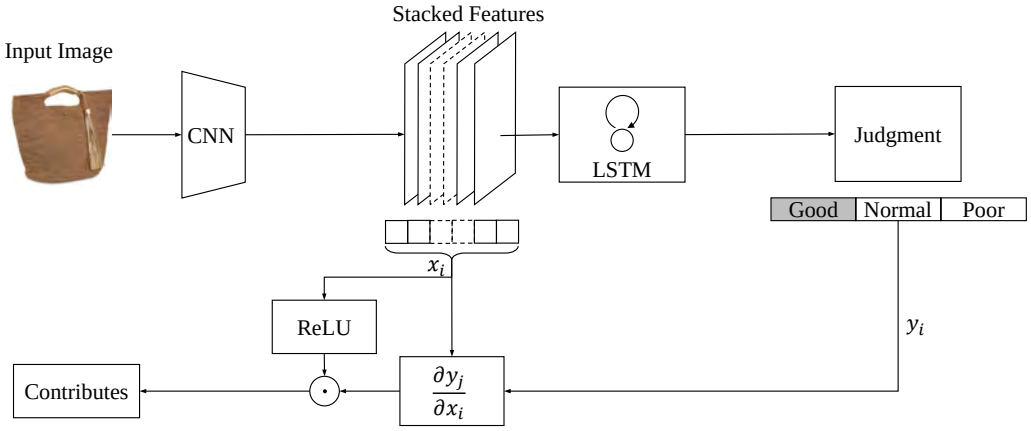


Figure 5.2 Overview of the Inter-factor Compatibility Network. Given the compatibility output computed by Bi-LSTM and the stacked features as input, the contribution of each element is determined based on the decision of judgment. This is achieved by performing a point-wise multiplication between the stacked feature and its backpropagation gradients.

5.3.3 Gradient Penalty Architecture

Normally, the neuron importance weight α_k^c [112] is defined as:

$$\alpha_k^c := \frac{1}{Z} \sum_i \sum_j \frac{\partial y^c}{\partial A_{ij}^k} \tag{5.6}$$

where i, j iterates over the spatial dimensions and Z is the number of pixels in the feature map. A weighted product of forward activation maps A^k is performed. Subsequently, a ReLU layer is performed on the weighted sum to obtain the heatmap $H_c^{\text{Grad-CAM}}$ as follows:

$$H_c^{\text{Grad-CAM}} := \text{ReLU}(\sum_k \alpha_k^c A^k) \tag{5.7}$$

The gradient penalty is exploited to predict the reason for judgment. Figure 5.2 depicts the details of the Inter-factor Compatibility Network. The

contribution of each element, denoted as contrib_j, is leveraged for the decision of judgment, which is calculated by:

$$\text{contrib}_j(\mathbf{x}_i) := \frac{\partial y_i}{\partial \mathbf{x}_i} \odot \text{ReLU}(\mathbf{x}_i) \tag{5.8}$$

where y_i is the logit for the judgment $j \in \mathcal{J}$, and $\mathbf{x}_i$ is one element of compatibility feature $\mathbf{x}_i \in \mathbf{x}$. The *positive contribution* of $\mathbf{x}_i$ for the judgment is calculated using the following equation:

$$\text{contrib}_j^+(r) := \frac{1}{|I_r|} \sum_{i \in I_r} \text{ReLU}(\frac{\partial y_i}{\partial \mathbf{x}_i}) \odot \text{ReLU}(\mathbf{x}_i) \tag{5.9}$$

where I_r is the index set of neurons for factor $r \in \mathcal{R}$.

The network is trained with specially designed regularizations so that the main reason predicted by the network is aligned with pre-labeled data. Cross-entropy regularizer is used to compute the reason loss as follows:

$$F_r := \sum_{j \in \mathcal{J}} \mathbb{I}_{j^{gt}}(j) \cdot \text{contrib}_j^+(r) - \text{contrib}_{\text{normal}}^+(r) \tag{5.10}$$

$$L_{\text{reason}} = -\log(\frac{\exp(F_{r^{gt}})}{\sum_{r \in R} \exp(F_r)}) \tag{5.11}$$

where $\mathbb{I}_{j^gt}$ is an indicator function for ground-truth judgment. If judgment j is the same as ground-truth label, $\mathbb{I}_{j^gt} = 1$; else, $\mathbb{I}_{j^gt} = 0$.

The total loss L is described as $L = L_{\text{judgment}} + \alpha L_{\text{reason}}$, where α is a hyper-parameter that is used to control the effect of reason regularization. As indicated in the definition of contribution (Equation 5.9) and reason (Equation 5.10), the Bi-LSTM network and inter-factor network are jointly trained because the loss term L_{reason} penalizes the gradient and the gradient penalty directly affects the network parameters.

5.4 Experiments

Experimental settings are first introduced in this section. Then, the effectiveness of the proposed approach is examined from the qualitative and quantitative aspects, including a detailed ablation study. Lastly, a demo website based on this research is developed.

5.4.1 Experimental Settings

Evaluation Datasets. The dataset used for evaluating models is the expanded EVALUATION3 dataset to address two limitations exhibited by the EVALUATION3 dataset [169]. Firstly, outfits in the EVALUATION3 dataset only contain top and bottom items. To this end, a bag and a pair of shoes are added to each outfit. All outfits are manually annotated from scratch because the original annotations for compatibility and reason are inappropriate for the expanded outfits. Secondly, it lacks attribute annotations for shoes and bags. Fashion experts are invited to annotate these added shoes and bags to address this limitation. The taxonomy of fashion attributes is presented in Table 5.1. To summarize, the expanded EVALUATION3 dataset contains 34,479 outfits splitting into 29,479 for training, 3,000 for validation, and 2,000 for testing. Each outfit has a corresponding judgment label and a reason label.

Parameter Settings. The embedding feature for each attribute is learned using the pre-trained *Resnet-18* models, which is optimized using the Adam method with an initial learning rate of 0.001 and a weight decay of 5×10^{-5}. As for the Bi-LSTM, the SGD optimization method is employed with an initial learning rate of 0.001 and weight decay of 5×10^{-4} for 140 epochs. The learning rate is decreased by a factor of ten after 84 epochs. In learning the corresponding reason, the regularization is applied in the form of cross-entropy.

Table 5.2 Comparison of different methods on the updated EVALUATION3 test set. All the evaluating experiments are repeated six times, and the values after ± are the mean square error.

Methods	reason accuracy
Multi-CLS-Part	74.8 ± 3.1
IFIV [130]	35.9 ± 4.5
Reason linear [169]	68.3 ± 2.4
Reason square [169]	73.8 ± 1.6
Reason cross-entropy [169]	76.7 ± 3.6
Bi-LSTM (Ours)	**77.6 ± 3.9**

5.4.2 Quantitative Analysis

The *reason accuracy* is used as the evaluation metric, which is defined as the ratio of correctly predicted reasons to the number of predicted judgments. The proposed method is compared with five baselines methods. 1). *Multi-CLS-Part* method separately train the compatibility judgment and reason using a standard multi-task classification model. 2). *Item Feature Influence Value* (IFIV) [130] method determine the explanation through the output scores of the item-feature pairs, and it is trained in an unsupervised manner. 3). Reason linear, square, and cross-entropy are the methods introduced in [169], where the form of reason regularization is linear, square, and cross-entropy, respectively.

Table 5.2 reports the evaluation results on the updated EVALUATION3 test set. The results show that the proposed Bi-LSTM method achieves the highest reason accuracy (77.6 ± 3.9), outperforming the other methods. The *Multi-CLS-Part* method achieves a reasonable accuracy (74.8 ± 3.1), while the IFIV method has a lower accuracy (35.9 ± 4.5). The reason linear, square, and cross-entropy methods perform comparably with accuracies of 68.3 ± 2.4, 73.8 ± 1.6, and 76.7 ± 3.6, respectively. These results demonstrate the effective-

Judgment ground truth: Good

Reason ground truth: Print

Color	**Print**	Top & Bottom	Bag	Shoes
0.06636	**0.08392**	0.00017	0.00014	0.00031

(a)

Judgment ground truth: Poor

Reason ground truth: Color

Color	Print	Top & Bottom	Bag	Shoes
0.23770	0.00002	-0.0154	-0.0001	0.00007

(b)

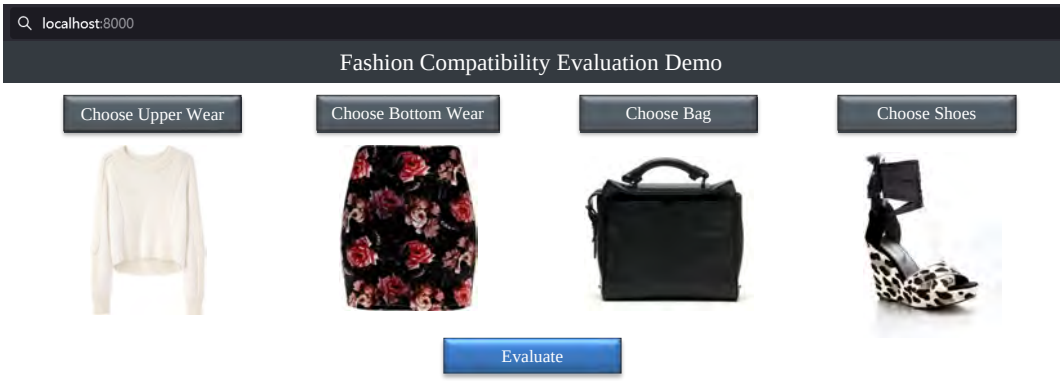
Figure 5.3 The table displays the contribution values of various candidate reasons, with the maximum value highlighted in bold.

ness of incorporating a bidirectional LSTM architecture in capturing intricate relationships and dependencies between fashion items and their features.

5.4.3 Qualitative Analysis

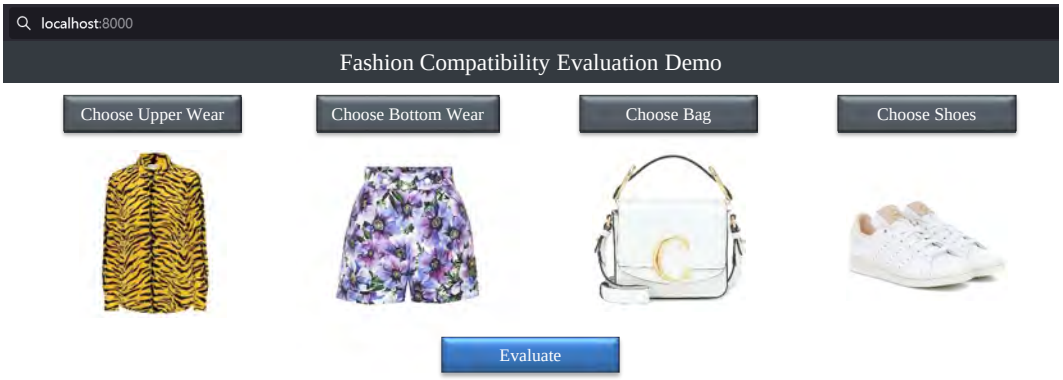
Figure 5.3 showcases the qualitative evaluation results obtained from the expanded EVALUATION3 dataset. In Figure 5.3 (a), the ground truth judgment and reason for the outfit are identified as *Good* and *Print*, respectively. The corresponding table presents the model's reason contribution scores, highlighting the highest score for *print* in bold. This numerical evidence establishes the model's consistency with the ground truth.

Similarly, for the outfit depicted in Figure 5.3 (b), the ground truth compatibility is labeled as *Poor* due to the mismatched orange coat and red pants. The predicted scores align with this observation, notably revealing a substantial disparity between the values of **0.238** for color and **0.00002** for print. This significant numerical difference attests to the method's confidence in its reason prediction.



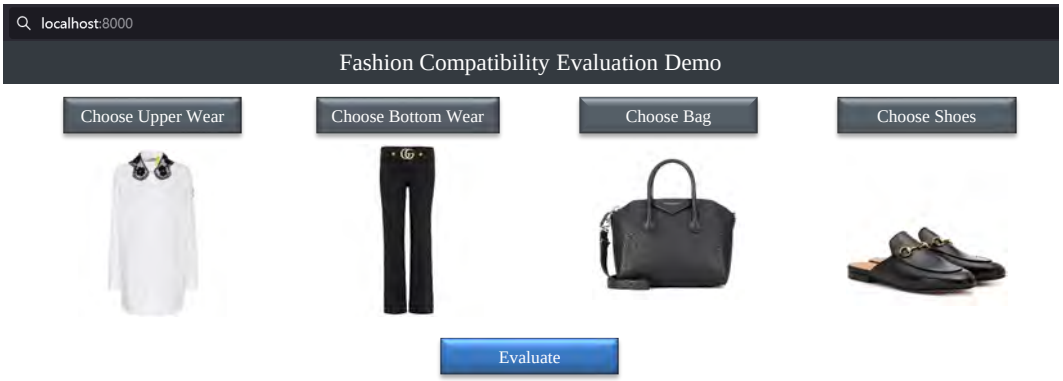
- Judgment: Good
- Explanation: The *plain print* top and the *floral print* bottom make the outfit in a novel style.

(a)



- Judgment: Poor
- Explanation: The *yellow* at the top and *blue* at the bottom are wrong color matching.

(b)



- Judgment: Normal
- Explanation: No specific explanation for a normal outfit.

(c)

Figure 5.4 A website demo application is developed that can predict the compatibility of an outfit and provide the corresponding explanation.

Website Demo. A website application based on the proposed model is developed. Figure 5.4 shows screenshots of three outfit evaluations. The operation process of this website is as follows:

1. Users are required to upload pictures of the top, bottom, bag, and shoes.

2. After they click the *Evaluate* button, the webpage will present the evaluation results of the outfit's compatibility and the corresponding explanation.

For the outfit shown in Figure 5.4 (a), the outfit is predicted to be *Good*, and the explanation states: *The plain print top and the floral print bottom create a novel style for the outfit.* The explanation is generated using a predesigned sentence template incorporating the predicted reason. Furthermore, the attribute values used in the explanation template, such as the print types in this example (*plain* for the top and *floral* for the bottom), are recognized using the feature extraction network.

Similarly, the outfit shown in Figure 5.4 (b) is predicted as *Poor* with an explanation as *The yellow at the top and blue at the bottom are wrong color matching.* The color name used for generating an explanation is initially extracted by the FOCO system in the HSB (Hue, Saturation, Brightness) form. Subsequently, the extracted main color vector undergoes a transformation into a color name that humans can understand.

As stated at the beginning of Section 5.3, there will be no specific explanation for this outfit for a *Normal* outfit. This website also maintains the same setting, as illustrated in Figure 5.4 (c), the presented outfit is evaluated as *Normal*, and the explanation section shows: *No specific explanation for a normal outfit.*

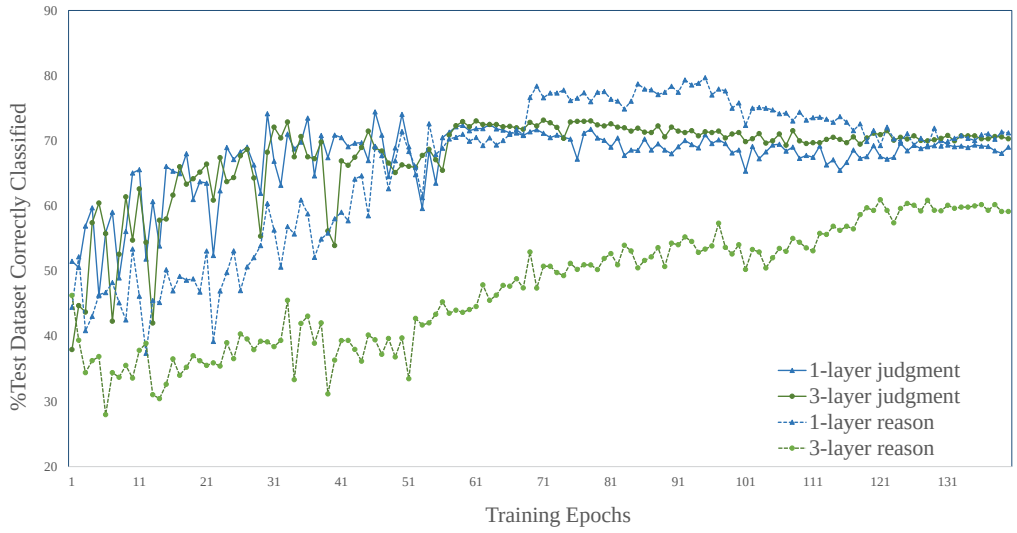


Figure 5.5 Learning curves of *judgment accuracy* and *reason accuracy* on the test set for Bi-LSTM with a different number of layers.

5.4.4 Ablation study

Number of LSTM layers. A series of experiments are conducted to investigate the effect of the number of layers of the LSTM network on evaluation accuracy. As shown in Figure 5.5, from the perspective of *judgment accuracy*, LSTM with three layers tends to get slightly better results. However, it is also observed that the *reason accuracy* of LSTM with three layers converges to 60% at the training time of 130 epochs. In contrast, *reason accuracy* of LSTM with one layer almost reaches 80% and gradually decreases with the increasing training time. Based on this result, one-layer LSTM is adopted as the architecture of the proposed model.

Dimension of LSTM hidden layer. The impact of the LSTM hidden layer's dimensionality on the model's performance is investigated. The model is trained six times with identical parameters, and the average accuracy values

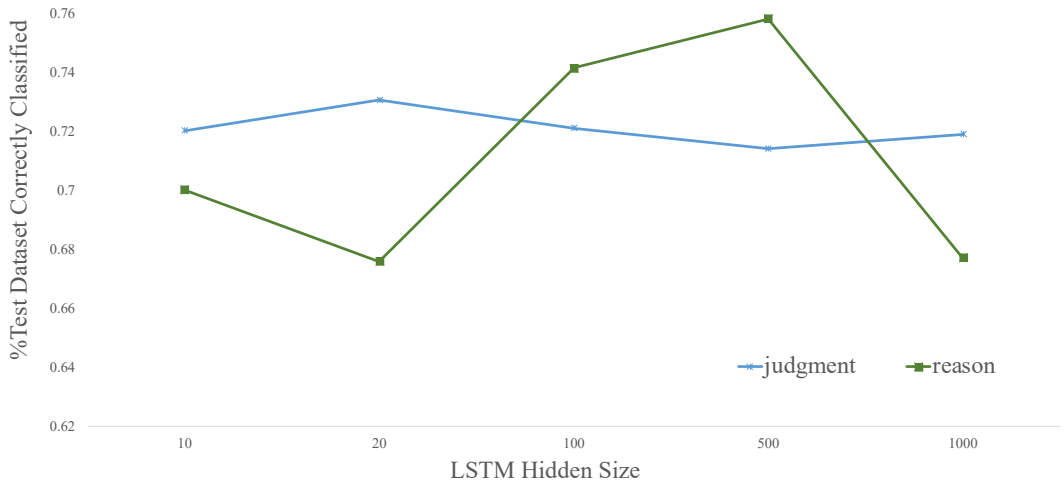


Figure 5.6 Curves of *judgment accuracy* and *reason accuracy* according to different dimensions of the LSTM hidden layer. Experiments include five different dimension conditions, which are 10, 20, 100, 500, and 1000. All the evaluating experiments are repeated six times on the test dataset.

on the test dataset are computed. Figure 5.6 illustrates the results, indicating that the size of the hidden layers has minimal effect on *judgment accuracy* but significantly influences *reason accuracy*. The highest *reason accuracy* of **0.76** is achieved when the size of the hidden layers is set to 500, which is selected as the hyperparameter for the proposed model.

5.5 Chapter Summary

In conclusion, this chapter presents a novel fashion compatibility model that combines the Bi-LSTM model and inter-factor compatibility network through joint training. The Bi-LSTM model is utilized to predict outfit compatibility by treating attribute features as a sequential input. The inclusion of gradient penalty regularization ensures that the generated explanations align with expert-annotated reasons.

The comprehensive evaluation, both qualitative and quantitative, demonstrates the effectiveness of the proposed system in accurately predicting outfit compatibility and providing corresponding reasons. Quantitative analysis reveals that the approach outperforms other methods in terms of *reason accuracy* on the updated EVALUATION3 dataset. Moreover, the numerical results showcase the model's ability to classify reasons based on the Bi-LSTM output and item features.

Finally, the chapter introduces a website application that utilizes the trained model, enabling users to upload outfit images and receive compatibility judgments along with explanations. This application serves as a practical demonstration of the proposed system's capabilities.

Chapter 6

Dress Well via Fashion Cognition Learning

6.1 Introduction

Fashion exists [94] in our daily life as a tool for expressing attitude and presenting culture. The convergence of fashion and artificial intelligence has garnered significant attention among researchers. While numerous studies have investigated the fashion recommendation problem, their practical application in real-world scenarios still presents considerable challenges. One limitation of existing models is that they primarily focus on evaluating the compatibility between fashion items without considering the overall compatibility between the outfit and individual customers during online shopping. As shown in Figure 6.1 (a), different customers have varied appearances, such as heights, hairstyles, and skin colors, directly affecting whether an outfit is compatible with them. For example, the outfit shown in Figure 6.1 (b) consists of a long white dress that is inappropriate to recommend to the second customer since

(a) Different customers with varied appearance

(b) Precise Outfit Recommendation with fashion cognitive model

Figure 6.1 Illustration of Fashion Cognition Learning task. The physical attributes of customers should be considered in fashion recommendations.

she is not so high enough to wear this long dress. Thus, even though this outfit is perfectly matched, it is inappropriate to recommend it to her.

Previous research mainly focused on the relations among fashion items via fashion compatibility learning [41, 71, 143, 161]. Many of them [72, 169] also focused on the explainability of fashion compatibility models. In addition, a few works noticed the influence of personal information, such as user preference [8, 78, 95], social media posts [123, 165], body shape [48]. However, no prior approach systematically considered the compatibility relationships between fashion items in an outfit and the varied appearance of online shoppers.

To address the above limitations, this chapter aims to provide a precise and appropriate fashion recommendation service to customers by considering their personal physical information. To distinguish from previous works utilizing the user's personal preference for personalized recommendation, a new task is

defined, namely **Fashion Cognition Learning**, that focuses on the influence of personal physical information on the compatibility of an outfit. This task is treated as a multi-label classification task. An end-to-end framework, Fashion Convolutional Network (FCN), is proposed, which learns the compatible relationships between outfits and humans. The FCN contains two modules: outfit encoder and Multi-label Graph Convolutional Networks (ML-GCN). The outfit encoder utilizes convolutional filters with different window sizes to encode the outfit into an outfit embedding. Applying filters with different sizes enables convolutional kernels to see different combinations of fashion attribute features. The ML-GCN is employed to learn multi-label classifiers based on word embeddings of physical labels. The predicted scores for all labels are obtained by multiplying classifier vectors with outfit embeddings.

Meanwhile, to facilitate the development of the proposed framework, a new outfit dataset is introduced covering personal physical information, namely Outfits for You (O4U). The O4U focuses on women's wear since women are the largest market among all types of crowd [3], and all labels are designed according to women's characteristics. It includes a total of 29,352 outfits. Each outfit is associated with two labels: 1). Whether the outfit is good or not; 2). Which kind of physical label is incompatible with the outfit. Six fashion experts are invited to label these outfits. The labeling procedure is carefully designed to maintain annotation consistency. Extensive experiments on the O4U dataset show that the proposed FCN outperforms other baselines.

6.2 Related Work

Personalization plays a crucial role in online selling services [3]. Previous research has focused on recommending items based on user preferences, leveraging various sources such as purchasing records and social media posts from platforms like Instagram [8, 66, 78, 84, 95, 111, 123, 165]. For instance, Packer *et al.* [95] developed an approach that models individual users' visual preferences using interpretable image representations, allowing for personalized clothing recommendations. Wen *et al.* [144] constructed knowledge graphs to capture correlations between clothing and context attributes, enabling personalized recommendations through the Apriori algorithm. Chen *et al.* [8] employed a Transformer architecture to connect user preferences with individual items and outfits. Zheng *et al.* [165] presented an item-to-set metric learning framework that learns to compute the similarity between a set of historical fashion items of a user to a new fashion item. Kim *et al.* [63] proposed a knowledge distillation framework for outfit recommendation, leveraging false-negative information from a teacher model without requiring the ranking of all candidates.

Some prior studies also investigated the task of dressing for diverse body shapes. Hidayati *et al.* [48] explored the compatibility of clothing styles and body shapes via a set of celebrities' photos. Hsiao *et al.* [52] introduced a visual body-aware embedding to capture the affinity between clothing items and different body shapes. However, their focus was solely on body shape, but they overlooked other physical characteristics. In light of this, this chapter tackles a novel task: learning the compatibility between outfits and diverse personal physical information.

Table 6.1 Details of personal physical features and sub-features.

Features	**Sub-features** (N - numbers of sub-features)
Body Shape	rectangle, top hourglass, athletics, round, spoon... (10)
Skin Color	yellow, dark, fair, brown (4)
Hair Style	long curls, long straight hair... (6)
Hair Color	ginger, black, dark brown, light brown... (6)
Height	high, middle, low (3)
Breasts Size	big, average, small (3)
Color-contrast	high, low (2)

6.3 O4U Dataset

Fashion cognition learning is based on fashion compatibility learning. It further learns the compatibility between outfits and personal physical features. Thus, the Outfit for You (O4U) dataset is built following the same structure of fashion compatibility learning.

Firstly, a labeling system for what types of personal body information may affect compatibility with garments is devised with reference to the current practices of fashion participants. The details of the defined label system are summarized in Table 6.1.

Secondly, to ensure the objectiveness of the created outfits to the maximum extent, 50,000 seed outfits are randomly generated. Each seed outfit consists of at least clothing items covering the whole body, one bag, one pair of shoes, and n accessories ($n \in [0, 5]$). Six experts majoring in fashion are invited to label those outfits. Determining if an outfit is well-matched is the first step. If true, they will select which personal features are incompatible with this outfit. Otherwise, this outfit only has a label to indicate that it is not well-matched. An outfit is only kept if the consistent accuracy of these six experts is over 95% in all 34 labels. The voting mechanism decides the few inconsistent annotation

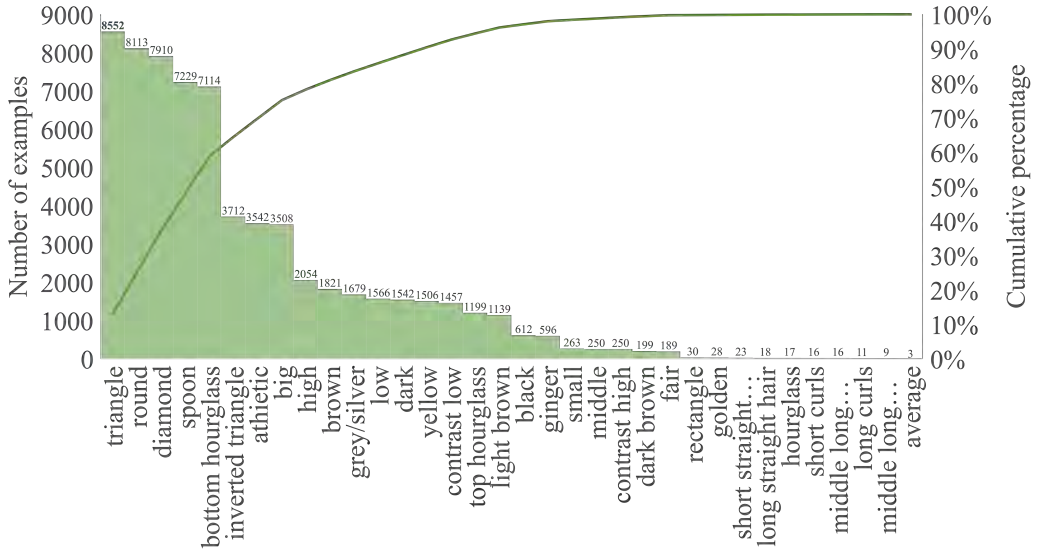


Figure 6.2 Number of examples for each physical label

results.

Finally, after the labeling process, there are 29,352 outfits retained. Meanwhile, only 15,748 outfits are labeled as well-matched, and the average unmatched physical label of these well-matched outfits is 5.25. The dataset is randomly divided into a training set, validation set, and test set in the form of 8:1:1. The label distribution of the training set is shown in Figure 6.2.

6.4 Approach

6.4.1 Problem Formulation and Motivation

The newly introduced task is formulated as a multi-label classification task to recognize whether the given outfit is compatible with multiple personal physical labels. Specifically, given a set of items $\mathcal{M} = \{p_i\}_i^{N_p}$ of N_p individual items and a collection $\mathcal{T} = \{O_j\}_{j=1}^{N_t}$ containing N_t outfits, each outfit $O =$

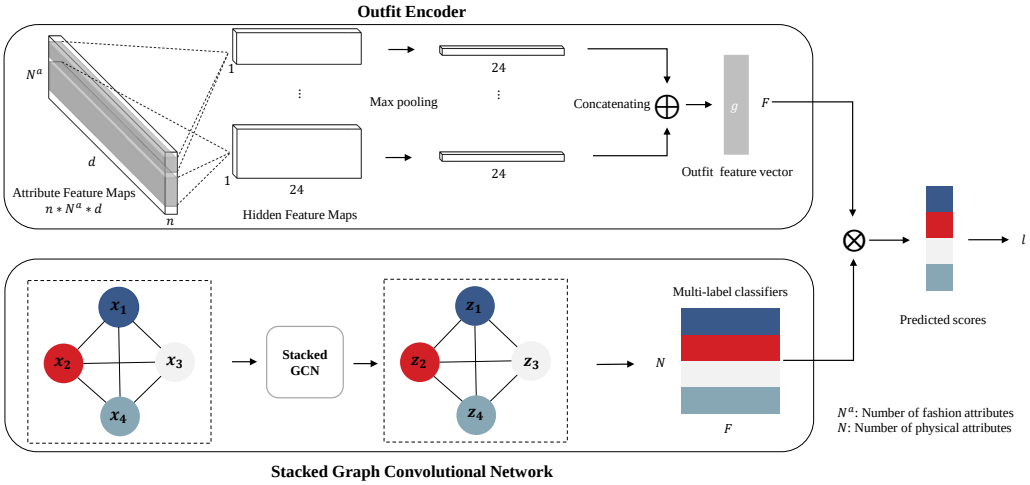


Figure 6.3 An overview of the proposed Fashion Convolutional Network. It comprises an outfit encoder and a stacked graph convolutional network. The outfit encoder is utilized for encoding outfits into outfit feature vectors by applying convolutional operation on attribute features. The stacked graph convolutional network is exploited to represent the classifiers of physical labels. Each physical label is treated as a node of the graph. The predicted scores are obtained by applying these label classifiers to the outfit feature vector.

$\{p_i\}_i^n$ in collection $\mathcal{T}$ is defined as a subset of $\mathcal{M}$ containing n different items. Each outfit O has a fashion compatibility label $l_f \in \{0, 1\}$ indicating whether this outfit is well-matched or not and a set of personal physical labels $\mathbf{l_p} \in \mathbb{R}^N$, where N is the number of physical labels defined in Table 6.1. Each item $p_i \in \mathcal{M}$ has its corresponding image I_i (unstructured data) and other metadata such as the primary color data, the category label, and some attribute labels $\mathbf{l_a}$ (structured data).

This research proposes the Fashion Convolutional Network (FCN) to address this task. As shown in Figure 6.3, FCN contains two modules: outfit encoder and stacked graph convolutional networks. In the outfit encoder, 1-dimensional convolutional filters of different sizes are proposed to extract the hidden features of the outfit. Two key motivations drive the design of a con-

volutional structure for encoding outfits. Firstly, fashion data exhibits translation invariance, whereby the ordering of items or attributes within an outfit does not impact its representation. This characteristic makes convolutional models well-suited for learning from such data. Secondly, the compatibility of an outfit with a physical label is contingent upon one or more fashion attributes. Convolutional filters of different sizes are strategically employed to aggregate different numbers of attribute features to capture these attributes accurately. This approach ensures an effective encoding of outfits and facilitates the extraction of meaningful fashion representations.

6.4.2 Outfit Encoder

The outfit encoder consists of a set of convolutional filters utilized to encode an outfit into an outfit embedding. Different from using a convolutional neural network (CNN) to extract features from item images, the proposed outfit encoder is applied to fashion attribute features. Given an outfit, fashion attribute features $\mathbf{X} \in \mathbb{R}^{N_a \times d}$ is extracted from each item using well pre-trained *Visual Geometry Group* (VGG) [120] network on a large-scale fashion attribute dataset, where N_a is the number of fashion attributes and d is the dimensionality of attribute features. Each attribute feature vector is the output of the last convolution layer after a max-pooling operation. These attribute features, serving as the input of the outfit encoder, are fixed during the whole training process. The advantage of using these attribute features compared to raw images is that the network can focus on the important features of items and make the training process more efficient.

Attribute feature maps of n items are presented by stacking (padded where necessary) all attribute features along the item dimension:

$$\mathbf{Z} = \mathbf{X}_1 \oplus \mathbf{X}_2 \oplus \cdots \oplus \mathbf{X}_n \tag{6.1}$$

where $\mathbf{Z} \in \mathbb{R}^{n \times N_a \times d}$ and $\oplus$ is the stacking operation. A convolutional layer contains N_c convolutional filters with different window sizes, and each filter has multiple convolutional kernels. The notation $\mathbf{w}_j \in \mathbb{R}^{h_j \times d}$ refers to j-th filters in this layer, where h_j means the filter is applied to a window of h_j attribute features to generate a new feature. The number of input and output channels of each filter is n and 24, respectively. The convolution stride and padding are fixed to 1 and 0, respectively. After the convolutional process, a max-pooling layer along the filter moving dimension is applied, yielding a 24-dimensional vector for each filter. The final outfit embedding, denoted as $\mathbf{g} \in \mathbb{R}^F$, is obtained by concatenating these convolved vectors, where F is the dimensionality of the outfit embedding.

6.4.3 Multi-label Graph Convolutional Networks

The Multi-label Graph Convolutional Networks [11] (ML-GCN) is used to train classifiers of the physical labels. ML-GCN is a graph convolutional networks (GCN) [64] based model, taking the advantage of capturing the label correlations by treating the classifiers of labels as nodes. As illustrated in Figure 6.4, the adjacency matrix $\mathbf{A}$ is constructed based on the conditional probability of label L_j when label L_i appears. The i, j entry of the matrix $\mathbf{A}$ is $\mathbf{A}_{ij} = P(L_j|L_i)$, and matrix $\mathbf{A}$ is a weighted and asymmetrical matrix.

The generic layer-wise propagation rule of a GCN layer is:

$$\mathbf{x}^{(l+1)} = \sigma(\widetilde{\mathbf{D}}^{-\frac{1}{2}}\widetilde{\mathbf{A}}\widetilde{\mathbf{D}}^{-\frac{1}{2}}\mathbf{x}^{(l)}\Theta^{(l)}) \tag{6.2}$$

where $\widetilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}_D$ is the adjacency matrix of the graph with self-connections and $\widetilde{\mathbf{D}} = \sum_j \widetilde{\mathbf{A}}_{ij}$ is the degree matrix. $\mathbf{A} \in \mathbb{R}^{N \times N}$ is the adjacency matrix where N denotes the number of nodes in the graph. $\mathbf{x}^{(l)} \in \mathbb{R}^{N \times C^{(l)}}$ is the

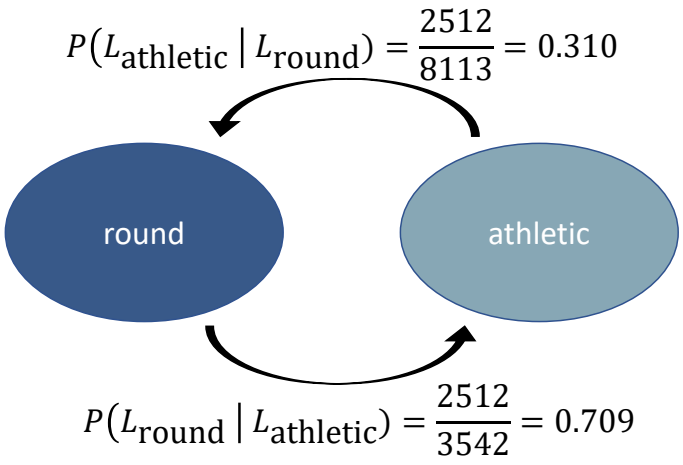


Figure 6.4 Construction of adjacency matrix based on the conditional probability of *round* and *athletic*. When *athletic* is not compatible with an outfit, there is a high probability that *round* is also not compatible with this outfit.

matrix of activations in the l^{th} layer with $C^{(l)}$ feature maps. $\Theta^{(l)} \in \mathbb{R}^{C^{(l)} \times C^{(l+1)}}$ is the trainable weight matrix. $\sigma(\cdot)$ denotes the nonlinear activation function. $\mathbf{x}^{(l+1)} \in \mathbb{R}^{N \times C^{(l+1)}}$ is the convolved feature matrix with $C^{(l+1)}$ feature maps.

A two-layer stacked GCN is selected to learn classifiers using the layer-wise propagation rule of Equation 6.2. Taking the label representation with C physical labels $\mathbf{X} \in \mathbb{R}^{N \times C}$ and the adjacency matrix $\mathbf{A} \in \mathbb{R}^{N \times N}$ as input, a two-layer GCN model $f(\mathbf{X}, \mathbf{A})$ can be expressed mathematically as:

$$\mathbf{Z} = f(\mathbf{X}, \mathbf{A}) = \widehat{\mathbf{A}}\ \mathrm{ReLU}(\widehat{\mathbf{A}}\mathbf{X}\mathbf{W}^{(0)})\mathbf{W}^{(1)} \tag{6.3}$$

where $\widehat{\mathbf{A}} = \widetilde{\mathbf{D}}^{-\frac{1}{2}}\widetilde{\mathbf{A}}\widetilde{\mathbf{D}}^{-\frac{1}{2}}$ is normalized version of adjacency matrix. $\mathbf{W}^{(0)} \in \mathbb{R}^{C \times H}$ and $\mathbf{W}^{(1)} \in \mathbb{R}^{H \times F}$ are two trainable weight matrices for the first and second layer, respectively and H is the dimension of the hidden layer. $\mathbf{Z} \in \mathbb{R}^{N \times F}$ is the classifier matrix with F feature maps.

By applying label classifiers $\mathbf{Z}$ to the outfit embedding $\mathbf{g} \in \mathbb{R}^{F}$, the pre-

dicted score $\widehat{\mathbf{y}}$ is a non-parametric product of them:

$$\widehat{\mathbf{y}} = \mathbf{Z} \cdot \mathbf{g} \tag{6.4}$$

The multi-label classification loss is computed as follows:

$$L_1 = \sum_{n=1}^{N} \mathbf{y}^n \log(\sigma(\widehat{\mathbf{y}}^n)) + (1 - \mathbf{y}^n) \log(1 - \sigma(\widehat{\mathbf{y}}^n)) \tag{6.5}$$

where $\mathbf{y} \in \mathbb{R}^N$ is the ground truth physical labels of an outfit, and $\sigma(\cdot)$ is the sigmoid function. The overall cost function is defined as follows:

$$J(\Theta_{\text{FCN}}) = L_1 + \frac{\lambda}{2} \|\Theta_{\text{FCN}}\|_2^2 \tag{6.6}$$

where Θ_{FCN} is the trainable parameters of FCN and λ is the L2 regularization hyperparameter.

6.5 Experiments

6.5.1 Experimental Settings

Implementation Details. For the outfit encoder, the convolutional layer has five filters with different window sizes, *i.e.*, 1, 2, 4, 6, and 8. The number of fashion attributes N_a is 14. For the GCN module, a two-layer stacked GCN is used, and the output dimension of these two layers are 200 and 120. A pre-trained VGG [120] is utilized as the attribute feature extractor. The images' height and width are cropped to 224, and the dimension of the attribute feature vectors is 512. The main color feature extracted using FOCO system [171] is also added to the input feature maps. The physical labels are encoded by

Glove [103] into 100-dimensional word embeddings. The FCN is trained in an end-to-end manner on the O4U dataset with a batch size of 10 on NVIDIA RTX 3070 GPU. The *Stochastic gradient descent* [108] algorithm is employed as the optimizer with the learning rate, momentum, and weight decay are $1e^{-1}$, 0.9, and 5e-5, respectively. An exponentially decreasing schedule for the learning rate and an early stop training strategy are adopted. The O4U dataset introduced in this work is used for the model evaluation since the existing datasets are inappropriate for the task of modeling fashion cognition.

Compared Approaches. 1). SVM [102]: The support vector machine (SVM) is chosen as one of the baselines to demonstrate the effectiveness of our approach. **2). Linear:** A network consists of multiple fully connected layers and ReLU activation functions. **3). ResNet [44]:** The ResNet is retrained by inputting the mean value of all item images. **4). Attention [135]:** Several stacked multi-head attention layers are stacked to encode an outfit with various attribute vectors into one vector.

6.5.2 Quantitative Results

Following the general practice [11, 132, 139], the performance of models on these metrics are reported, including mean average precision (mAP); average per-class precision (CP), recall (CR), and F1 (CF1); average overall precision (OP), recall (OR), and F1 (OF1). Average per-class metrics evaluate each label individually and then average over all labels. Average overall metrics evaluate over all examples. The results of these metrics on top-3 labels are also reported.

The mAP results for the body shape attributes are reported in Tables 6.2, while the mAP results for the resting physical attributes are reported in Ta-

Table 6.2 Quantitative results on Body Shape attributes.

Methods	top hourglass	hourglass	athletics	inverted triangle	triangle	spoon	round	dimension
Linear	**15.47**	63.90	66.47	76.14	63.41	63.09	71.96	70.90
ResNet [44]	10.73	31.76	33.37	79.22	67.86	67.15	66.44	65.26
Attn [135]	9.48	30.20	31.69	69.68	59.07	57.46	61.36	61.52
FCN (proposed)	15.39	**66.53**	**70.15**	**83.48**	**70.29**	**69.82**	**77.52**	**76.35**

Table 6.3 Quantitative results on the physical attributes excluding body shapes.

Methods	Skin			Hair color		Height		Breasts	Contrast
	yellow	dark	brown	light brown	grey	high	low	big	low
Linear	11.59	24.35	14.35	9.17	13.31	17.38	13.94	31.23	12.27
ResNet [44]	12.05	41.57	14.29	**9.83**	**13.68**	18.08	11.98	26.50	12.06
Attn [135]	12.31	11.68	14.27	8.42	12.24	14.30	12.43	27.51	12.29
FCN	**13.24**	**46.84**	**15.11**	9.31	13.00	**21.57**	**23.23**	**31.91**	**12.72**

ble 6.3. The proposed method FCN achieves the best performance over 14 out of 17 labels compared with other baseline methods. Especially on labels belonging to the body shape category, FCN achieves a considerable improvement compared to other methods.

Model performances covering all 17 labels are reported in Table 6.4. FCN outperforms other baselines on almost all metrics. SVM, an effective machine learning method, performs well regarding average overall metrics. However,

Table 6.4 Quantitative results on main metrics.

Methods	All							Top-3					
	mAP	CP	CR	CF1	OP	OR	OF1	CP	CR	CF1	OP	OR	OF1
SVM [102]	-	28.07	33.10	30.38	68.70	61.54	64.90	-	-	-	-	-	-
Linear	37.59	26.59	**33.93**	29.81	63.23	**65.14**	64.17	28.96	20.57	24.06	68.25	41.29	51.46
ResNet	34.22	22.83	27.55	24.97	64.29	57.18	60.53	23.98	18.80	21.08	67.52	40.06	50.29
Attn [135]	29.76	18.18	29.41	22.47	61.82	62.33	62.07	11.44	17.65	13.88	64.82	39.22	48.87
FCN	**42.14**	**32.29**	33.84	**33.04**	**68.89**	62.17	**65.36**	**34.16**	**21.06**	**26.06**	**73.25**	**41.32**	**52.83**

FCN surpasses SVM by 4.22, 0.74, and 2.66 on the CP, CR, and OP. The linear method works best in the recall indexes, indicating that this method may have a high sensitivity to the labels. The performance of ResNet is not good on mAP, and it indicates that treating outfits as the mean value of item images is not a good idea for this task.

Query 1

Ground Truth	Proposed FCN	SVM	Attention	Linear	ResNet
Inverted triangle	Inverted triangle	Top hourglass	Triangle	Inverted triangle	Triangle
Triangle	Triangle	Hourglass	Round	Round	Round
Round	Round	Athletic	Diamon	Diamon	Diamon
Diamon	Diamon		Rectangle		
			Bottom hourglass		

Query 2

Ground Truth	Proposed FCN	SVM	Attention	Linear	ResNet
Triangle	Triangle	Triangle	Triangle	Round	Triangle
Spoon	Spoon	Spoon	Spoon	Diamon	Spoon
Round	Round	Bottom hourglass	Round	Inverted triangle	Round
Diamon	Diamon	Athletic	Diamon		Diamon
		Inverted triangle	Bottom hourglass		Bottom hourglass

Query 3

Ground Truth	Proposed FCN	SVM	Attention	Linear	ResNet
Triangle	Triangle	Bottom hourglass	Triangle	Round	Triangle
Round	Round	Spoon	Round	Inverted triangle	Round
			Spoon	Diamon	Diamon
			Diamon		Spoon
			Bottom hourglass		Bottom hourglass

Figure 6.5 Qualitative results of baseline methods and the proposed FCN. The text in red is the wrong prediction. FCN precisely predicts all incompatible body shapes for the query outfit.

6.5.3 Qualitative Results

Figure 6.5 presents the qualitative results conducted on the O4U dataset. For the first query outfit, the ground truth column reveals that individuals with *Inverted triangle*, *Triangle*, *Round*, and *Diamond* body figures are unsuitable for this outfit. This unsuitability arises from the mismatch between the tank top's silhouette and the straight-line pants concerning these specific body shapes. Remarkably, the FCN accurately identifies all incompatible body shapes, surpassing the performance of other comparative methods. Similarly, for the other two outfits, the FCN method precisely predicts the mismatched body shapes, whereas other baseline methods provide incorrect judgments.

6.5.4 Ablation Study

Effect of filter region size. The sensitivity of different combinations of filter region size is explored. As shown in Table 6.5, only using one convolutional filter size shows the worst performance. Using filters with a big region size (relative to attribute number 14) harms model performance. Using multiple filters with the same size achieves the best result on mAP and OF1, but the results are lower than FCN on the top-3 labels. The combination used in FCN (1, 2, 4, 6, 8) shows the best performance on CF1 and Top-3 metrics.

Table 6.5 Effect of filter region size.

Region size	All							Top-3					
	mAP	CP	CR	CF1	OP	OR	OF1	CP	CR	CF1	OP	OR	OF1
(1)	40.68	32.55	32.99	32.77	67.68	62.50	64.99	29.56	20.27	24.05	71.29	40.53	51.67
(2)	38.93	28.36	32.42	30.25	68.67	61.28	64.77	30.64	20.44	24.52	73.01	40.24	51.89
(4,4,4,4,4)	**43.11**	**32.82**	33.46	**33.13**	68.70	62.02	65.19	32.30	20.82	25.32	72.30	40.87	52.22
(8,9,10)	41.38	28.29	32.72	30.35	68.83	61.29	64.85	30.29	21.19	24.93	73.12	40.96	52.51
(1,2,4,6,8)	42.14	32.29	**33.84**	33.04	**68.89**	**62.17**	**65.36**	**34.16**	**21.06**	**26.06**	**73.25**	**41.32**	**52.83**

Table 6.6 Effect of numbers of kernels for each filter.

# Kernels	All							Top-3					
	mAP	CP	CR	CF1	OP	OR	OF1	CP	CR	CF1	OP	OR	OF1
2	35.46	26.54	35.58	30.40	62.52	68.25	65.26	27.94	19.72	23.12	66.03	39.95	49.78
12	41.67	32.19	33.91	33.03	67.96	63.09	65.44	**36.09**	20.77	**26.37**	72.31	40.95	52.29
24	42.14	32.29	**33.84**	**33.04**	68.89	**62.17**	**65.36**	34.16	**21.06**	26.06	73.25	**41.32**	**52.83**
48	**42.65**	**32.55**	33.10	32.82	**69.17**	60.90	64.77	34.75	21.02	26.20	73.75	40.78	52.52

Table 6.7 Effect of numbers of GCN layers.

# GCN	All							Top-3					
	mAP	CP	CR	CF1	OP	OR	OF1	CP	CR	CF1	OP	OR	OF1
1	40.85	**34.11**	32.09	**33.07**	68.19	61.38	64.61	25.01	18.93	21.55	71.63	39.67	51.06
2	**42.14**	32.29	**33.84**	33.04	68.89	**62.17**	**65.36**	**34.16**	**21.06**	**26.06**	**73.25**	**41.32**	**52.83**
4	40.73	28.59	32.24	30.31	**69.05**	60.46	64.47	30.49	20.83	24.75	73.02	40.40	52.02
8	39.45	28.00	32.29	29.99	67.73	60.19	63.74	21.25	19.55	20.36	71.18	35.54	47.41

Effect of numbers of kernels for each filter The effect of the different numbers of kernels is also explored. The filter region size is kept the same, and the results are reported in Table 6.6. It can be observed that the performance achieves the best results when the number of kernels is 24. Using too few convolutional kernels will deteriorate performance significantly. Using too many kernels cannot dramatically improve performance, and it hurts recall metrics.

Effect of numbers of GCN layers Finally, the effects of different numbers of GCN layers is examined and reported in Table 6.7. The results show that deeper multi-layer GCNs degrade the performance on almost all metrics. Therefore a two-layer stacked GCN is chosen in FCN.

6.6 Chapter Summary

In summary, this chapter introduces the task of Fashion Cognition Learning, which aims to learn the compatibility between fashion outfits and personal physical information. To tackle this task, the Fashion Convolutional Network framework is proposed, which utilizes visual-semantic embeddings of outfit composition and appearance features of individuals to capture the relationships. Additionally, a large-scale fashion outfit dataset is constructed, encompassing comprehensive personal physical information. The extensive experimental results demonstrate the superior performance of the proposed framework compared to alternative methods. This research contributes to the advancement of understanding and modeling the intricate connections between fashion outfits and individual characteristics, paving the way for personalized fashion recommendations and improved compatibility assessment.

Chapter 7

Learning Body-shape-Aware Embeddings for Fashion Compatibility

7.1 Introduction

Fashion Recommendation Systems (FRSs) [8, 55] is not a new topic, but they still have great potential for economic benefits. Previous works have mainly focused on fashion compatibility learning (FCL) [20, 71, 97], which only considers the compatibility among fashion items. However, in addition to the outfit itself, consumers are highly concerned about its appearance when worn. Figure 7.1 demonstrates how fashion compatibility can vary depending on different body shapes. For instance, individuals with an *inverted triangle* body shape may find the outfit in Figure 7.1 (a) suitable, while those with a *triangle* body shape may not.

To effectively incorporate accurate body shape information into FRSs, it

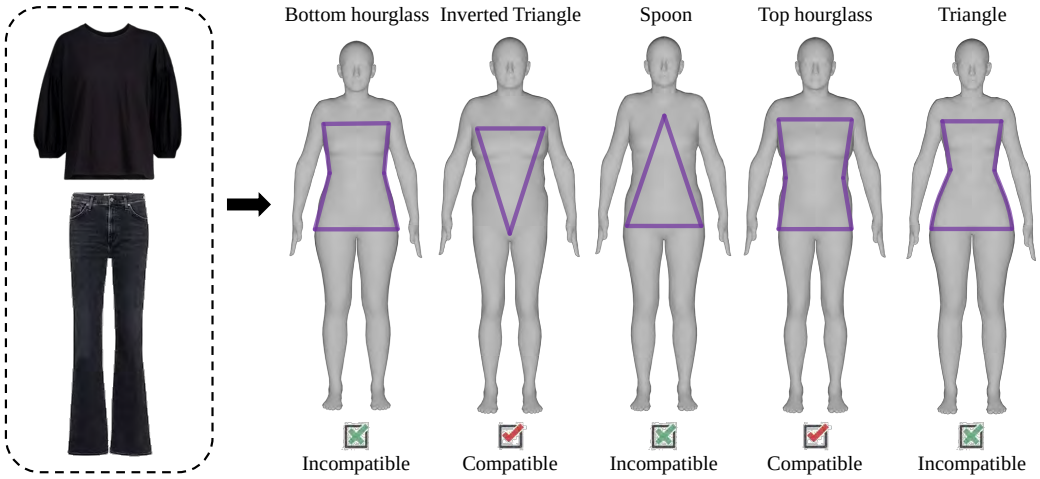


Figure 7.1 An example of the body-shape-aware fashion compatibility task. The outfit of a shirt and a pair of pants is compatible with the *inverted triangle* and *top hourglass* body shapes but incompatible with others.

is essential to leverage valuable information from body images. However, previous studies [47, 48] merely rely on body measurement data, overlooking the valuable visual features of body shape, which limits their ability to provide precise recommendations. Moreover, accurately representing outfits is also critical, as the scaling and spatial relationships between clothing items can impact how they fit and flatter different body shapes. Therefore, conventional outfit representation methods used in FCL, such as item-wise correlations [17, 134, 151] or graph neural networks [18, 124], are insufficient for modeling the relationships between body shape and an outfit. Lastly, providing a reasonable explanation for the evaluation is crucial for personalized FRSs. However, previous studies [47, 90, 98] have not achieved this.

To this end, this paper proposes a Body-shape-Aware Network (BA-Net) to model the relationships between body shape and outfit. BA-Net consists of three modules: Body-shape Embedding Module (BEM), Outfit Embedding

Module (OEM), and Joint Embedding Module (JEM). The BEM combines visual and anthropometric features to obtain a general representation of the body shape. However, obtaining accurate visual features from body images requires a diverse dataset with explicit body shape annotations, which is currently unavailable. Thus, a new dataset is created covering five common body shapes; each contains 4,000 3D body models with varying but similar shapes. Every body model in the dataset is accompanied by its anthropometric data and frontal view image. The OEM learns the outfit embedding by incorporating visual and textual features of the outfit. For the visual features, the try-on appearance of an outfit is leveraged because it contains the scaling and spatial relationships among individual clothing items. A Multi-layer Try-on System (M-VTON) is developed to generate the realistic try-on image of an outfit. For the textual aspect, the fashion attributes information is exploited to enhance the outfit representation, where the attribute values are encoded into word embeddings. Finally, the JEM integrates body shape and outfit representations to compute the body-shape-aware embedding, which is then transformed by a linear function to obtain the final compatibility score. The core of the OEM and JEM is a cross-modal attention mechanism that allows them to merge features from different modalities. The hierarchical design of BA-Net facilitates the propagation of cross-modal interactions between fashion attributes and body shapes through the computed attention maps. These attention maps are utilized to generate the attribute-level explanations for the prediction results. The proposed BA-Net is compared with the state-of-the-art methods on the O4U dataset introduced in Chapter 6. Both qualitative and quantitative results show the advancement of the BA-Net.

7.2 Related Work

7.2.1 Body-shape-Aware Fashion Compatibility.

With the development of FCL, researchers are increasingly aware of the importance of body shape to practical applications. The primary challenge of this task lies in accurately encoding and classifying human body shapes. Hidayati *et al.* [48] represent the body shape using body measurements collected from websites. They employed an affinity propagation [31] algorithm to cluster the measurement data into several body shapes. Sun *et al.* [126] proposed to use 3D features consisting of 240 vectors to represent female upper body shapes. Pang *et al.* [98] employed a GCN to learn body shape representations based on statistical correlations between physical labels. Simmons *et al.* [118] developed a well-known body shape classification system called the Female Figure Identification Technique (FFIT), which uses anthropometric data measured from 3D body scans for classification. Subsequent research [22, 101, 155] improved the FFIT, which has become a widely accepted standard for body shape classification. However, these approaches neglect the informative images of human bodies. In this chapter, the body shape is encoded into a more comprehensive embedding incorporating both anthropometric features and visual features, which are extracted from body images based on the newly introduced body shape dataset.

7.2.2 Outfit Representation

Most research tried to learn item representations rather than outfit representations in modeling fashion compatibility. Vasileva *et al.* [134] modeled fashion compatibility by measuring item similarities respecting item types. Cucu-

rull *et al.* [17] proposed to generate item embeddings by considering product context. Tan *et al.* [128] claimed to learn item embeddings without explicit supervision to alleviate the deficiency of rich labeling costs. The graph-based method OCM-CF proposed in [124] utilized a multi-head attention mechanism to leverage the contextual information of fashion items and obtain outfit representations. Pang *et al.* [98] encoded the outfit through 1-dimensional convolutional filters with different sizes based on the features of fashion attributes. Hidayati *et al.* [47] utilized photos of female celebrities, and they embedded the outfit through the separate items segmented from these photos.

The main limitation of these approaches in encoding the outfit is that they omit the scaling and spatial relationships between individual clothing items. To address the limitation, some studies proposed representing outfits using try-on appearances to enhance the model performance. Dong *et al.* [25] developed a Multi-modal Try-on Template Generator (MTTG), which explores both visual and textual modalities of fashion items. Zheng *et al.* [166] utilized a teacher-student knowledge distillation scheme, where the teacher network is trained through unsupervised self-encoding, enabling the student network to accurately represent try-on outfits by imitating the teacher's output and deriving the representation directly from the discrete clothing items. However, these approaches still suffer from poor quality of try-on images due to loss of garment details, such as print and texture, which are crucial for the fashion recommendation task. In light of this, a new Virtual try-on framework is proposed in this chapter, which can produce try-on results reliably and expressly.

7.3 Body Shape Dataset

Previous studies [48, 101] have introduced few body shape datasets, but the numbers of body models they contained are insufficient to represent body shapes. For example, Parker *et al.* [101] analyzed 1,679 3D body scans, while only 10 and 62 human bodies are categorized as *triangle* and *top hourglass* body shapes, respectively. The dataset introduced by Hidayati *et al.* [48] consists of 3,150 individual celebrities with their body measurements, while no body shape labels are provided. In light of this, a new dataset is constructed for the body shape representation. The definition of body shape is first given. Let $\mathbf{\Omega} = \{\mathbf{T}_i\}_{i=1}^{N_T}$ be a set of human body models, where $\mathbf{T}_i$ represent a 3D body model. A body shape set $\mathbf{U}^k \subseteq \mathbf{\Omega}$ is defined as a subset of $\mathbf{\Omega}$ whose models share similar characteristics and can be categorized into the same body shape, where $k \in [1, K]$ and K is the number of examined body shapes. The newly introduced dataset covers **five** common body shapes, namely *bottom hourglass, inverted triangle, spoon, top hourglass*, and *triangle*. Each body shape set $\mathbf{U}^k$ contains 4,000 human body models. The detailed construction process includes the following steps:

Step 1: Generating SMPL Model. 200,000 3D body models are generated with diverse body sizes to ensure dataset variety by employing the Skinned Multi-Person Linear (SMPL) model [82]. SMPL is a learned model that accurately represents various human body sizes in different poses. The body model is generated according to shape parameters $\boldsymbol{\beta}$ and pose parameters $\boldsymbol{\theta}$. However, as this task focuses on variations in human body shape, the pose parameters are kept constant while generating body models. Thus, a one-to-one correspondence exists between the body model set and the shape parameter

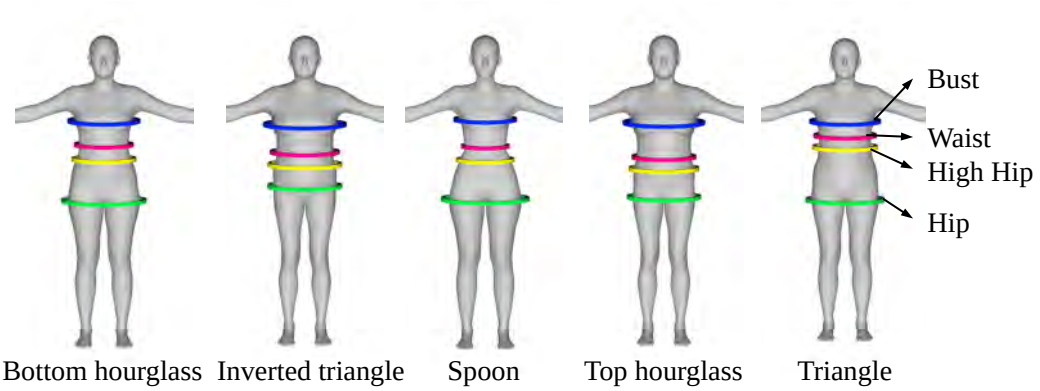


Figure 7.2 Five common body shapes included in the newly introduced body shape dataset. Both *Bottom hourglass* and *top hourglass* have a well-defined waistline, but their difference lies in their hip-to-bust ratio; *triangle* and *inverted triangle* lack a well-defined waistline because they do not consider the bust-to-waist ratio; *Spoon* is characterized by a large gap between hip and bust circumference and a smaller bust-to-waist ratio than the *hourglass*.

set. Let $\mathcal{F}_{\text{SMPL}}$ be the SMPL [82] model forward function, the body model is denoted as $\mathbf{T}_i = \mathcal{F}_{\text{SMPL}}(\boldsymbol{\beta}_i)$.

Step 2. Measuring Anthropometric Data. A body measurement tool [113] is employed to acquire the anthropometric data containing 20 dimensions from the generated 3D model. Among these 20 measures, the *bust*, *waist*, *high hip* and *hip* circumferences are most important because FFIT [155] is employed to identify the body shape based on these four measures. Figure 7.2 visualizes these circumferences, where the circle's size indicates the circumference's length. These circumferences are measured by locating body landmarks based on the regularities of cross-sectional body shapes. The tool's localization criteria is modified to maintain consistency with the body landmarks defined in FFIT. The obtained anthropometric data is denoted as $\boldsymbol{\omega} = \mathcal{F}_{\text{measure}}(\mathbf{T})$, where $\mathcal{F}_{\text{measure}}$ is the measuring process.

Step 3. Cleaning Invalid Model. To ensure the generated body models are realistic, invalid models that fall outside the standard range of human height-weight distribution [58, 88] are eliminated. Consequently, 11.57% of the generated body models are retained. Then, the mean and variance of the remaining models' shape parameters $\boldsymbol{\beta}$ are calculated. Using these distributions, a new set of 100,000 body models are generated. This process improves the realism of the newly generated bodies, as the height and weight are more closely aligned with the normal distribution of humans.

Step 4. Annotating Body Shape. The FFIT algorithm [155] is utilized to determine the body shape of each SMPL model. However, the classification results show that the distribution of body models across different body shapes is non-uniform. For instance, out of the 100,000 body models, only a small portion is identified as the *top hourglass* and *triangle*, with only 120 and 11 models, respectively. To address this imbalance issue, its specific shape parameter distributions are computed for each body shape, which is then used to regenerate body models. This method effectively enhances the occurrence frequencies of these underrepresented body shapes by optimizing the shape parameters. Finally, 4,000 valid human body models for five body shapes (as illustrated in Figure 7.2) are randomly sampled to form the dataset.

Step 5: Capturing Frontal View Image. A frontal view image of each body model is captured by rendering it in a virtual environment using an orthographic camera. The resulting image has a resolution of 1024×512 pixels and is saved in PNG format. The notation $\mathcal{F}_{\text{ortho}}$ represents the orthographic projection process, and the resulting image is denoted as $\mathbf{I} = \mathcal{F}_{\text{ortho}}(\mathbf{T})$.

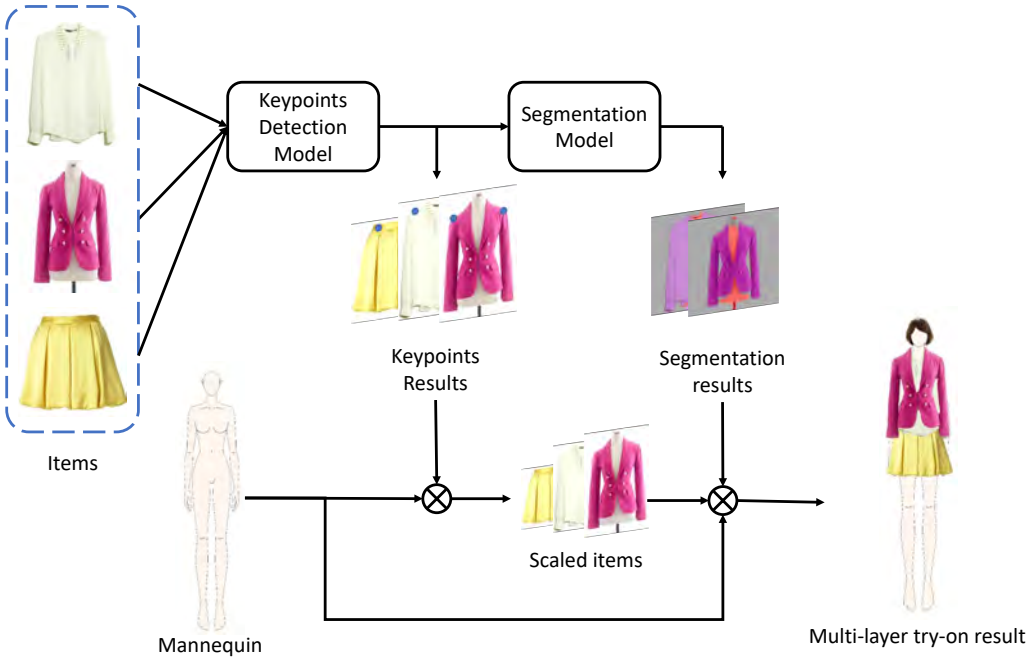


Figure 7.3 Overview of the multi-layer try-on system. Each fashion item is first sent to keypoint detection model to obtain fashion keypoints. The estimated keypoints are used to calculate scales and positions for garments. Meanwhile, the segmentation model separates item images into front and back pieces. Finally, the segmentation results are applied to scaled items to generate the try-on images.

7.4 Multi-layer Virtual Try-on System

The proposed M-VTON consists of two modules, as shown in Figure 7.3: keypoint detection model and the clothing segmentation model. Both are based on deep learning methods.

7.4.1 Keypoint Detection

The keypoint detection task aims to detect K keypoints from an image. In the context of fashion, detecting fashion-oriented keypoints for garments is the primary objective. It is worth noting that fashion keypoint detection and human pose estimation differ in some aspects, such as clothing being more difficult because of non-rigid deformations. There are two reasons why the

method ViPNAS [147] is employed, although it belongs to pose estimation. 1). Images in the mainstream fashion datasets are dominated by product images that suffer litter from non-rigid deformations. 2). One major drawback of fashion keypoint detection methods is inefficient.

The pose estimation model ViPNAS [147] is utilized. It can provide comparable performances with lower computation consumption by using Neural Architecture Search. ViPNAS contains two sub-models which are S-ViPNet aiming to estimate keypoints on key frames, and T-ViPNet aiming to estimate video-based keypoints. Since the goal is to extract keypoints from images instead of a video, only the S-ViPNet is used. Specifically, heatmaps are firstly regressed based on the high-resolution representation extracted by HRNetV1 [140]. The kth heatmap indicates the confidence of the location of kth keypoint.

7.4.2 Fashion Segmentation

A fashion segmentation model can parse out apparel's front and back pieces. However, existing datasets are inappropriate for training the desired segmentation model. Thus, a new dataset is constructed that focuses on the fashion layer's semantic segmentation, where details are reported in Table 7.1. Creating this dataset involves two steps. Firstly, 10,000 images of fashion items are collected, including authentic product images and sketch images, in a ratio

Table 7.1 Split details of segmentation dataset.

	Training	Validation	Testing
Real	1,992	87	87
Sketch	4,613	213	213
All	6,605	300	300

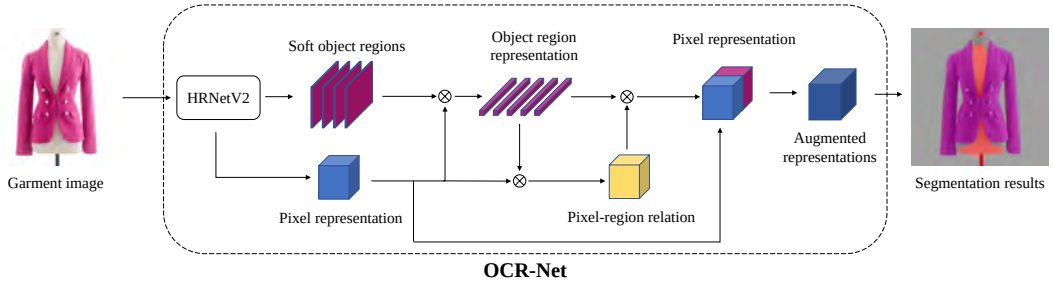


Figure 7.4 Illustration of the OCR-Net [158] model employed in M-VTON for fashion segmentation.

of 3:7. Secondly, data cleansing and pixel-level annotation are carried out. Finally, the data are split into training, validation, and testing sets.

OCR [157] (Object-Contextual Representations) is a popular segmentation method that fully utilizes all representations of object regions belonging to the corresponding class to augment one pixel's representation. Due to its advantage, OCR is employed for garment segmentation based on the newly created dataset. As shown in Figure 7.4, there are three steps of OCR:

(1). Soft object region. The HRNetV2 [140] is chosen as the backbone to segment K soft object regions from the given image I. Each object region $\mathbf{M}_k$ is a coarse segmentation result represented as a matrix. Each entry of $\mathbf{M}_k$ means the degree to which the pixel belongs to the corresponding class k.

(2). Representation of object region. The representation of each object region f_k is obtained using the following function:

$$\mathbf{f}_k = \sum_{i \in I} \tilde{m}_{ki} \mathbf{x}_i \tag{7.1}$$

where $\mathbf{x}_i$ is the representation of pixel p_i, and $\tilde{m}_{ki}$ represents the normalized degree that p_i belongs to k-th object region.

(3). Representation of pixel. The representation of pixel p_i is computed

after considering the relations between it and all object regions:

$$\mathbf{y}_i = \rho(\sum_{k=1}^{K} \omega_{ik}\delta(\mathbf{f}_k)) \tag{7.2}$$

where $\rho(\cdot)$ and $\delta(\cdot)$ are transformation functions. ω_{ik} indicates the relation between the pixel and object region:

$$\omega_{ik} = \frac{e^{\kappa(\mathbf{x}_i,\mathbf{f}_k)}}{\sum_{j=1}^{K} e^{\kappa(\mathbf{x}_i,\mathbf{f}_j)}} \tag{7.3}$$

where function $\kappa(\cdot)$ is an unnormalized relation function.

The final representation of the pixel p_i is obtained by aggregating $\mathbf{x}_i$ and $\mathbf{y}_i$:

$$\mathbf{z}_i = g([\mathbf{x}_i^T \mathbf{y}_i^T]^T) \tag{7.4}$$

where $g(\cdot)$ is the same transformation function as $\rho(\cdot)$. Finally, a pixel-level cross-entropy loss is applied to learn the segmentation model.

7.4.3 Outfit Synthesis

Outfit synthesis involves synthesizing separate clothing items into a cohesive try-on image. Firstly, the scale of each item is determined by aligning the keypoints of the garments with the corresponding keypoints on a mannequin to ensure proper fitting and proportions. Secondly, a layering system is implemented to represent the items in a multi-layered fashion. The segmentation results are utilized to divide the clothes into front and back pieces. Following a predefined try-on order, the scaled items are pasted onto the target positions piece by piece. The try-on order follows a sequence from outerwear, dress, blouse, skirt, to trousers, representing the outermost to the innermost layers of the outfit.

7.5 Body-shape-Aware Network

In this section, the proposed approach of learning body-shape-aware embeddings is elaborated. Specifically, the task formulation is first clarified. Then, the representations of body type, try-on image, and fashion attributes are given. Lastly, the architecture of BA-Net is described, containing three modules: Body-shape Embedding Module (BEM), Outfit Embedding Module (OEM), and Joint Embedding Module (JEM).

7.5.1 Task Formulation

Following [98], this task is formulated as a multi-label classification task. Given a training set $\mathcal{T} = \{O^j, Y^j\}_{j=1}^N$ containing N outfits, $O^j = \{\mathbf{X}^j, \mathbf{G}^j\}$ is denoted as the j-th outfit containing individual clothing images $\mathbf{X}^j$ and structured fashion attributes $\mathbf{G}^j$. $Y^j = \{y_k^j | k = 1, \cdots, K\}$ refers to a set of ground truth labels for j-th outfit conditioned on K body shapes, where $y_k^j = 1$ indicates that outfit O^j is **incompatible** with k-th body shape.

The goal is to devise a learning function $\mathcal{F}$ to predict the compatibility score $\hat{y}_k^j$ between a query outfit O^j and k-th body shape:

$$\hat{y}_k^j = \mathcal{F}(\{\mathbf{X}^j, \mathbf{G}^j, \boldsymbol{\omega}^k, \mathbf{I}^k\} | \boldsymbol{\Theta}) \tag{7.5}$$

where $\boldsymbol{\omega}^k$ and $\mathbf{I}^k$ are the anthropometric data and front view image of k-th body shape, respectively. $\boldsymbol{\Theta}$ is the training parameters.

7.5.2 Body-shape Representation

A Body-shape Embedding Module (BEM) is devised to compute the embedding for the body shape by exploiting both visual and anthropometric features

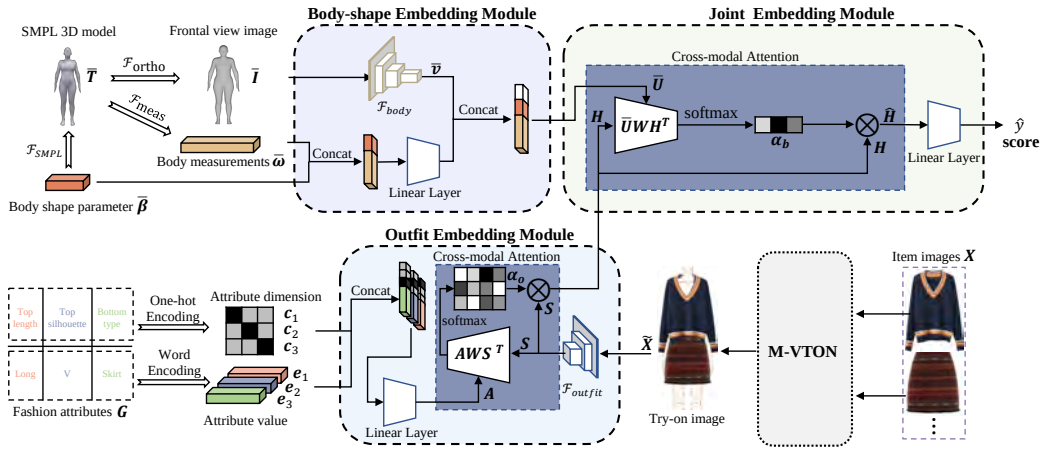


Figure 7.5 The proposed BA-Net consists of three modules. The Body-shape Embedding Module represents the body shape using both body image features and anthropometric features. In the Outfit Embedding Module, outfit visual features are extracted from the try-on image, which is synthesized via the Multi-layer Virtual Try-On Network (M-VTON). The fashion attribute features are merged with visual features to produce the outfit embedding through a cross-modal attention mechanism. Finally, both body shape features and outfit features are sent to the Joint Embedding Module to learn body-shape-aware embeddings. The cross-modal attention mechanism employed in BA-Net computes attention weights that can be used to generate attribute-level explanations for the prediction results.

extracted from a representative body model of this body shape, as illustrated in the top-left corner of Figure 7.5. The first step is to obtain the representative model for the k-th body shape. The shape parameters of all body models belonging to the set $\mathbf{U}^k$ are averaged, and then the SMPL model [82] generates the representative model according to the averaged parameters:

$$\bar{\mathbf{T}}^k = \mathcal{F}_{\text{SMPL}}(\bar{\boldsymbol{\beta}}^k) = \mathcal{F}_{\text{SMPL}}\Big(\frac{1}{|\mathbf{U}^k|}\sum_{\mathbf{T}_i \in \mathbf{U}^k} \boldsymbol{\beta}_i\Big) \tag{7.6}$$

where $\bar{\mathbf{T}}^k$ is the representative 3D model of k-th body shape, and $\bar{\boldsymbol{\beta}}^k \in \mathbb{R}^{1\times 10}$ is the averaged shape parameter vector. $|\mathbf{U}^k|$ means the size of set $\mathbf{U}^k$. Based on the representative model, an orthographic camera is used to capture the

corresponding frontal view image, denoted as $\bar{\mathbf{I}}^k = \mathcal{F}_{\text{ortho}}(\bar{\mathbf{T}}^k)$.

Visual features of k-th body shape are extracted from $\bar{\mathbf{I}}^k$ by employing a ResNet [44] model, which is trained on the body images of the introduced body shape dataset. This dataset is divided into the training (80%), validate (10%), and testing (10%) sets. The visual feature extraction process can be expressed as follows:

$$\bar{\mathbf{v}}^k = \mathcal{F}_{\text{body}}(\bar{\mathbf{I}}^k) \tag{7.7}$$

where $\bar{\mathbf{v}}^k \in \mathbb{R}^{1\times 512}$ is the visual features, and $\mathcal{F}_{\text{body}}$ refers to the forward function of ResNet with the last linear layer discarded.

The representative model is also measured to acquire the anthropometric data, denoted as $\bar{\boldsymbol{\omega}}^k = \mathcal{F}_{\text{measure}}(\bar{\mathbf{T}}^k) \in \mathbb{R}^{1\times 20}$, where $\mathcal{F}_{\text{measure}}$ refers to the measuring process. Since body shape parameters also contain information for characterizing the body shape, $\bar{\boldsymbol{\beta}}^k$ and $\bar{\boldsymbol{\omega}}^k$ are concatenated and then sent to a linear layer consisting of a linear transformation and a Rectified Linear Unit (ReLU) activation function. The resulting output is concatenated with $\bar{\mathbf{v}}^k$ to produce the body-shape embedding, denoted as $\bar{\mathbf{U}}^k \in \mathbb{R}^{1\times 1024}$. Formally, $\bar{\mathbf{U}}^k$ is calculated using the following equation:

$$\bar{\mathbf{U}}^k = \text{Concat}(\text{ReLU}(\text{Concat}(\bar{\boldsymbol{\beta}}^k, \bar{\boldsymbol{\omega}}^k)\mathbf{W}_B + \mathbf{b}_B), \bar{\mathbf{v}}^k) \tag{7.8}$$

where $\mathbf{W}_B \in \mathbb{R}^{30\times 512}$ and $\mathbf{b}_B \in \mathbb{R}^{1\times 512}$ are fully connected layer's weight matrix and bias vector, respectively. The resulting body shape features will be sent to the next module for joint embedding learning.

7.5.3 Try-on Image Representation

Try-on images are generated utilizing the Multi-layer Virtual Try-On Network (M-VTON), which can synthesize separate item images while preserving clothing details as much as possible. The generated try-on image, denoted as $\tilde{\mathbf{X}}$, has a resolution of 1040×680 pixels. A pre-trained ResNet [44] model is utilized to extract spatial features from the obtained try-on image. This model's last pooling layer and linear layer are discarded. The motivation behind encoding it into multiple region-level features is that they can provide more accurate representations than a single global feature. Formally, the feature extraction process can be expressed as:

$$\mathbf{S} = \mathcal{F}_{\text{outfit}}(\tilde{\mathbf{X}}) = \{\mathbf{x}_1, \cdots, \mathbf{x}_n\}; \tag{7.9}$$

where $\mathbf{S}$ is the representation of try-on image containing 128 spatial features $\mathbf{x}_i \in \mathbb{R}^{512}$, and $\mathcal{F}_{\text{outfit}}$ refers to the forward function of the modified ResNet.

7.5.4 Fashion Attributes Representation

Each clothing item in the O4U dataset is associated with a set of fashion attributes, which are manually recognized from various attribute dimensions. For the sake of explanation, three fashion attributes are shown in the bottom-left part of Figure 7.5. For example, the top item (on the right side of Figure 7.5) is categorized as *long* in terms of *top length* dimension of fashion attributes. The union of all attributes associated with each item in an outfit is assigned to represent the fashion attributes for the entire outfit.

For fashion attribute value, a pre-trained GloVe [103] model is employed to encode its text into a word embedding, denoted as $\mathbf{e} \in \mathbb{R}^{d_{\text{text}}}$, where $d_{\text{text}} = 300$

is the dimensionality of the word embedding. For fashion attribute dimension, it is encoded into a one-hot vector, denoted as $\mathbf{c} \in \mathbb{R}^{N_A}$, where $N_A = 15$ is the number of all fashion attributes used in this work. Then, $\mathbf{c}$ and $\mathbf{e}$ are concatenated to represent one fashion attribute and then apply a linear transformation to the concatenated vector. Suppose the j-th outfit possesses L^j fashion attributes, this outfit's attribute representation $\mathbf{A}^j \in \mathbb{R}^{L^j \times 512}$ is computed by:

$$\mathbf{A}^j = \{\text{ReLU}(\text{Concat}(\mathbf{c}_l, \mathbf{e}_l)\mathbf{W}_A + \mathbf{b}_A)\}_{l=1}^{L^j} \tag{7.10}$$

where $\mathbf{W}_A \in \mathbb{R}^{315 \times 512}$ and $\mathbf{b}_B \in \mathbb{R}^{512}$ is the weight matrix and bias vector of the linear transformation.

7.5.5 Body-type-Aware Network Architecture

The cross-modal attention block [85] is utilized in both the Outfit Embedding Module (OEM) and Joint Embedding Module (JEM) of BA-Net to merge data representations from different modalities. This mechanism improves conventional attention mechanisms by introducing a learnable weight matrix in the score function, where two modalities are connected by calculating their compatibility scores. Specifically, it takes two inputs denoted as a *query* $\mathbf{Q} \in \mathbb{R}^{N_q \times d_q}$ and a *value* $\mathbf{V} \in \mathbb{R}^{N_v \times d_v}$. The attention weights $\boldsymbol{\alpha} \in \mathbb{R}^{N_q \times N_v}$ is computed using the following equation:

$$\boldsymbol{\alpha} = \text{softmax}(\mathbf{Q}\mathbf{W}\mathbf{V}^T) \tag{7.11}$$

where $\mathbf{W} \in \mathbb{R}^{d_q \times d_v}$ is the learnable weight matrix, and the softmax operation is applied on the second dimension. According to the obtained attention

distribution and value $\mathbf{V}$, the output of this block is computed by:

$$\hat{\mathbf{V}} = \boldsymbol{\alpha}\mathbf{V} \tag{7.12}$$

where $\hat{\mathbf{V}} \in \mathbb{R}^{N_q \times d_v}$ is the fused feature vectors.

The OEM aims to acquire the outfit representation, denoted as $\mathbf{H}^j \in \mathbb{R}^{L^j \times 512}$, through integrating features of try-on image and fashion attributes using the cross-modal attention block:

$$\mathbf{H}^j = \boldsymbol{\alpha}_o \cdot \mathbf{S}^j = \text{softmax}(\mathbf{A}^j \mathbf{W}_o \mathbf{S}^{j^T}) \cdot \mathbf{S}^j \tag{7.13}$$

where $\mathbf{W}_o \in \mathbb{R}^{512 \times 512}$ is the learnable weight matrix and $\boldsymbol{\alpha}_o \in \mathbb{R}^{L^j \times 128}$ is the attention maps calculated in OEM. Then JEM learns the relationship between the k-th body shape features $\bar{\mathbf{U}}^k$ and the j-th outfit representation and outputs the compatibility vector between these two representations:

$$\hat{\mathbf{H}}_k^j = \boldsymbol{\alpha}_b \cdot \mathbf{H}^j = \text{softmax}(\bar{\mathbf{U}}^k \mathbf{W}_b \mathbf{H}^{j^T}) \cdot \mathbf{H}^j \tag{7.14}$$

where $\hat{\mathbf{H}}_k^j \in \mathbb{R}^{1 \times 512}$ is the body-shape-aware embedding, and $\mathbf{W}_b \in \mathbb{R}^{1024 \times 512}$ is the learnable weight matrix in the JEM. $\boldsymbol{\alpha}_b \in \mathbb{R}^{1 \times L^j}$ is the attention maps computed in JEM. It can be observed that the second dimension of $\boldsymbol{\alpha}_b$ is exactly the same as the number of fashion attributes associated with the j-th outfit. Based on this characteristic of the BA-Net, the corresponding explanations can be generated based on the influence distribution of fashion attributes reflected in the attention maps computed in JEM. $\boldsymbol{\alpha}_b$ is visualized in Section 7.6.4 to demonstrate the explainability possessed by BA-Net.

Lastly, the compatibility score is computed by applying a linear transfor-

mation on $\hat{\mathbf{H}}_k^j$:

$$\hat{y}_k^j = \hat{\mathbf{H}}_k^j \cdot \mathbf{W}_s + b_s \tag{7.15}$$

where $\mathbf{W}_s \in \mathbb{R}^{512\times 1}$ and b_s (scalar) denote the linear transformation's weights and bias, respectively. Since the task is formulated as a multi-label classification task, the binary cross entropy loss is used to measure the difference between predicted scores $\hat{y}_k^j$ and target scores y_k^j.

7.6 Experiments

This section aims to showcase the benefits of the proposed BA-Net model by addressing several research questions:

- **RQ1**: Does the BA-Net superior to the current state-of-the-art methods?
- **RQ2**: To what extent do the individual components of BA-Net influence the model's performance?
- **RQ3**: How does the BA-Net explain evaluation results?
- **RQ4**: How does the proposed model perform in the perceptual study?

7.6.1 Experimental Settings

Dataset. The proposed network is evaluated on the public dataset O4U, which contains 15,748 compatible outfits and 82,017 individual clothing items. Each item is associated with a product image and several fashion attributes. On average, the top item contains 6.64 fashion attributes, while the bottom item contains 3.77 attributes. This chapter focuses on learning the relationship between body shape and outfit. The body shape annotations are provided

by the O4U dataset covering five common body shapes. To ensure a fair comparison, the original training, validation, and testing data split provided by O4U is used.

Baselines. To demonstrate the superiority of the BA-Net, it is compared with five state-of-the-art methods:

(1). **StyleMe**[47], which extends AuxStyles[48] by using bidirectional symmetrical deep neural networks to learn a joint representation of outfits and body shapes, and measures the compatibility score with a cosine distance function. It is trained using separate item images and anthropometric data.

(2). **TDRG**[163], an effective multi-object recognition model that explores the structural and semantic aspect relations through Graph Convolutional Network. It learns the joint relation of the try-on image.

(3). **M3TR**[164], a multi-modal multi-label recognition model that effectively incorporates global visual context and linguistic information through ternary relationship learning. The body shape labels are embedded into the word embedding as the linguistic information and use try-on appearances as input images.

(4). **CSRA**[168], which captures spatial regions of objects from different categories by effectively combining a simple spatial attention score with class-specific and class-agnostic features. CSRA is trained using try-on images as input.

(5). **FCN**[98], which employs a convolutional layer to embed the outfit based on fashion attribute features, and utilizes a GCN to learn multi-label

classifiers based on word embeddings of body shapes. The compatibility scores are obtained by applying the learned classifiers to the outfit embedding.

Evaluation Metrics. Following the general practice [32, 90, 98], seven evaluation metrics are utilized to assess the performance of the proposed BA-Net compared with other methods. These metrics include mean average precision (mAP), average per-class precision (CP), recall (CR), F1 score (CF1), average overall precision (OP), recall (OR), and F1 score (OF1). Among these metrics, mAP, CF1, and OF1 are relatively more important, as they provide a more comprehensive assessment.

Implementation Details. The SGD optimizer [110] with momentum factor equalling 0.9 and weight decay 5e-4 is adopted. Moreover, the learning rate is gradually decreased according to the formula:

$$\text{lr} = \text{base\_lr} \times (1 - \text{step\_num}/\text{max\_step})^{0.9} \tag{7.16}$$

where the base learning rate is 0.1. The maximum steps and training batch size are set to 1,260 and 10, respectively. During the training process, the checkpoint model is saved at the highest mAP performance achieved on the validation set. The average evaluation results from five repeated experiments for all experiments are reported.

Table 7.2 Quantitative comparison among different methods.

Methods	mAP	CP	CR	CF1	OP	OR	OF1
Random	45.01	44.27	23.04	30.31	44.91	21.93	29.47
StyleMe [47]	49.08	37.50	56.05	44.94	62.81	77.70	69.47
TDRG [163]	54.66	50.80	63.60	56.48	65.42	78.85	71.51
M3TR [164]	61.37	55.92	61.19	58.44	69.37	79.65	74.15
CSRA [168]	61.38	56.63	61.18	58.82	71.82	76.79	74.22
FCN [98]	62.34	56.96	62.41	59.55	71.42	78.14	74.62
BA-Net (Ours)	**63.14**	**57.30**	**64.85**	**60.84**	**72.02**	**80.73**	**76.13**

7.6.2 Comparative Results (RQ1)

The quantitative and qualitative results of the BA-Net compared with state-of-the-art methods are reported to demonstrate the proposed approach's effectiveness.

Quantitative Results. Table 7.2 shows the quantitative results. All baseline methods are trained on the O4U training set. The random method means all predictions are given randomly. The proposed BA-Net achieves the best performances across all metrics. Specifically, it surpasses StyleMe by a clear margin (+14.06 on mAP). This may be because the bidirectional symmetrical deep neural networks utilized in StyleMe are limited in their ability to learn cross-modal relationships. Compared with the TDRG, M3TR, and CSRA methods, the BA-Net brings consistent +1.78˜8.5 mAP gains, +2.02˜4.36 CF1 gains, and +1.9˜4.6 OF1 gains over them. This may be attributed to BA-Net taking advantage of body shape features. BA-Net also outperforms the FCN method on all metrics. This may be attributed to the fact that FCN learns body shape representation from textual data. The quantitative analysis demonstrates the superiority of the proposed BA-Net in modeling the relationship between body shape and outfit.

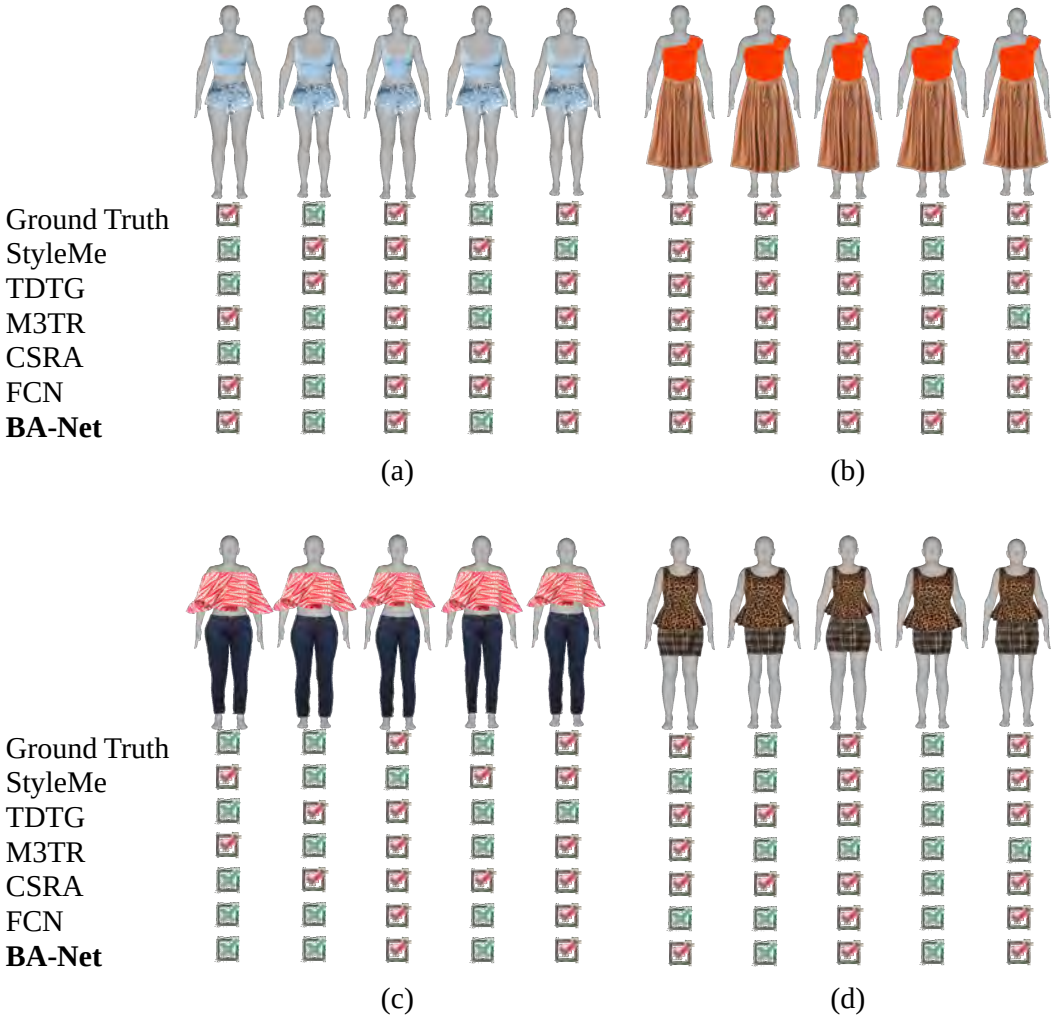


Figure 7.6 Qualitative comparison among different methods. The tick symbol indicates a match between the outfit and the body shape, while the cross symbol indicates a mismatch.

Qualitative Results. The quantitative results are presented in Figure 7.6. It is evident that among all baselines, the BA-Net consistently performs well with various outfit compositions. In Figure 7.6 (a), for example, the outfit consists of corset straps with hot pants, which might not be compatible with people having lower body segment obesity due to tight pants. However, the length of hot pants is short, exposing the legs, which can alleviate the feeling of envelopment, and thus, the outfit can still be compatible with body shapes such as *bottom hourglass, spoon*, and *triangle*. On the other hand, corset straps are heavy for people with *inverted triangle* or *top hourglass* body shapes, which also have big boobs. Thus, matching hot pants with the same large exposure of skin is incompatible. In contrast, as shown in Figure 7.6 (b), changing clothing to a tank top and A-line long skirt can solve both problems. Similarly, in Figure 7.6 (c), the off-shoulder blouse is unsuitable for people with broad shoulders, and the tight jeans are incompatible with those with lower body segment obesity. Furthermore, for outfits with special silhouettes, such as the peplum top with an H-line short skirt in Figure 7.6 (d), the BA-Net can still accurately assess the compatibility between body shape and the outfit.

7.6.3 Ablation Study (RQ2)

In this subsection, several ablation studies are conducted to examine the effectiveness of different components in the BA-Net, including representation learning, network structure, outfit encoding, and body shape features.

Ablation Study on Representation Learning. The effectiveness of three data representations used in BA-Net is investigated, including the body shapes, try-on images, and fashion attributes. The results of this ablation study are reported in Table 7.3. Firstly, the overall contribution of BA-Net to the multi-

Table 7.3 Ablation results on representation learning. *backbone*: utilizing backbone (ResNet-18) as multi-label classifier. *w/o-body*: encoding the body shape into one-hot vector. *w/o-try-on*: encoding outfit using visual features from separate items. *w/o-attr*: removing fashion attribute data.

Methods	mAP	CP	CR	CF1	OP	OR	OF1
backbone	57.71	54.47	57.54	55.96	67.53	76.39	71.68
w/o-body	60.57	55.68	60.71	57.97	67.46	73.84	70.46
w/o-try-on	61.72	56.29	62.77	59.35	71.43	78.99	75.02
w/o-attr	61.45	55.83	63.32	59.34	70.61	79.50	74.79
Full model	**63.14**	**57.30**	**64.85**	**60.84**	**72.02**	**80.73**	**76.13**

label classification performance is investigated by comparing it with BA-Net's *backbone* model (ResNet-18). The proposed full network brings +5.43 mAP, +4.88 CF1, and +4.45 OF1 performance improvements. Secondly, the effectiveness of the body-shape embedding method is assessed by removing body features and encoding body shape into a one-hot vector (*w/o-body*). The full model surpasses *w/o-body* by +2.57 mAP, +2.87 CF1, and 5.67 OF1. This result indicates the importance of visual and anthropometric body features in this task. Thirdly, the try-on embedding method is compared with a separate item embedding method (*w/o-try-on*). The result shows that BA-Net using the try-on embedding achieves higher scores (+1.42 mAP, +1.49 CF1, and +1.11 OF1) than the model using separate items. This result suggests that the proposed try-on embedding method captures more information from the try-on image compared with discrete items. Lastly, the impact of exploiting fashion attributes is examined in BA-Net. The results of *w/o-Attributes* demonstrate that utilizing fashion attribute data can improve the model's overall performance, with the full model achieving increases of +1.69 mAP, +1.50 CF1, and +1.34 OF1. These results suggest that fashion attributes can provide valuable cues for personalized fashion recommendation systems.

Table 7.4 Comparison on variations of cross-modal attention.

Structure	mAP	CP	CR	CF1	OP	OR	OF1
dot-product	50.02	39.36	42.70	40.96	65.56	58.89	62.05
multi-layer	52.30	52.63	25.47	34.33	52.63	28.36	36.86
multi-head	58.71	54.27	60.24	57.10	68.05	79.65	73.39
cross-modal	**63.14**	**57.30**	**64.85**	**60.84**	**72.02**	**80.73**	**76.13**

Ablation Study on Network Structure. Furthermore, three variations of the cross-modal attention mechanism are tested, and the quantitative results are reported in Table 7.4. Specifically, the cross-modal attention is replaced with the *dot-product* attention, *i.e.*, and the weight W is removed from Equation 7.11. The performance of the model is observed to decrease due to this operation. A possible reason may be attributed to the difference in input modalities, as the attention module in BA-Net receives inputs from different modalities, which is unsuitable for simple dot-product attention. Furthermore, the results of adopting *multi-layer* and *multi-head* of cross-modal attention are given. However, they fail to achieve better results.

Outfit Try-on Image Encoding Validation. Table 7.5 validates the usefulness of utilizing the try-on appearance of the outfit. This ablation study uses

Table 7.5 Comparison of encoding outfits with and without try-on images. The bold numbers indicate a larger value.

Method	Outfit Encoding	mAP	CP	CR	CF1	OP	OR	OF1
TDRG [163]	separate	49.97	37.54	56.61	45.14	62.78	78.44	69.74
	try-on	**54.66**	**50.80**	**63.60**	**56.48**	**65.42**	**78.85**	**71.51**
M3TR [164]	separate	53.90	52.99	57.07	54.95	64.34	78.13	70.57
	try-on	**61.37**	**55.92**	**61.19**	**58.44**	**69.37**	**79.65**	**74.15**
CSRA [168]	separate	57.38	55.92	54.59	55.24	69.31	72.81	71.02
	try-on	**61.38**	**56.63**	**61.18**	**58.82**	**71.82**	**76.79**	**74.22**

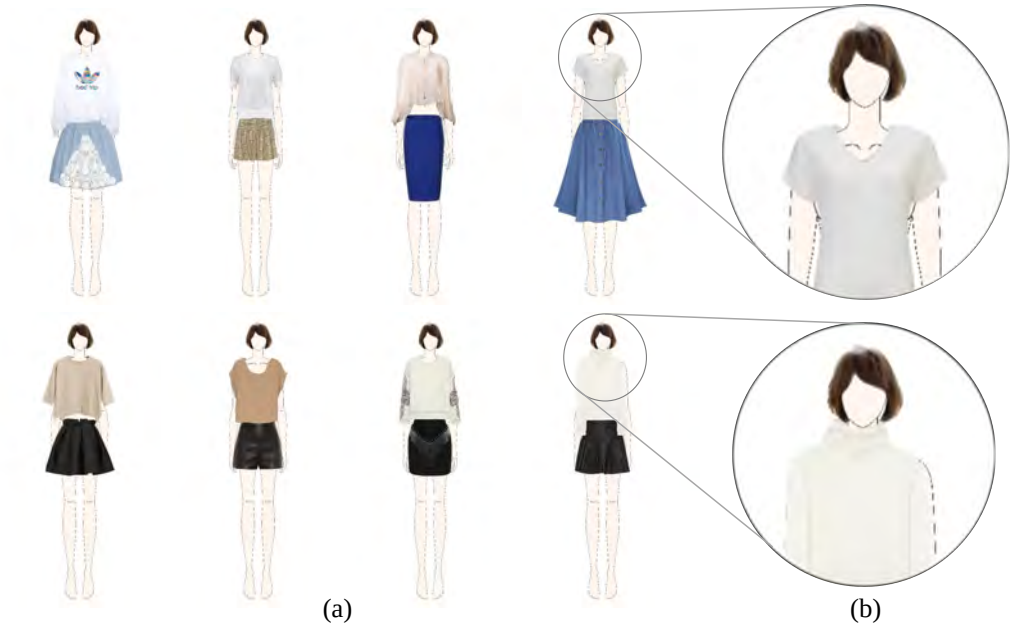


Figure 7.7 Try-on results of outfits from the O4U dataset.

three baseline methods by comparing their performances with try-on image encoding versus separate item encoding. From the table, it can be observed that performance consistently improved across all methods when using the try-on image to represent the outfit. Notably, M3TR achieves +7.47, 3.49, and 3.58 improvements on mAP, CF1, and OF1 metrics, respectively. This is reasonable to infer that owing to the try-on image capturing the interdependent relationships between clothing items, which allows M3TR to learn the underlying contextual relationships better.

The qualitative try-on results of eight outfits in O4U dataset are also presented in Figure 7.7 (a). It can be observed that garments are appropriately scaled and placed in the correct position. Figure 7.7 (b) shows a zoom-in view of the neckline position. M-VTON accurately depicts the interaction between the mannequin's neck and the top. More try-on results generated by the

Table 7.6 Body shape classification accuracy comparing with available classifiers.

Available body shape classifiers				Ours
Lee *et al.* [155]	Francis [30]	Collings [14]	Hidayati *et al.* [47]	
28.63%	31.84%	37.87%	76.83%	97.60%

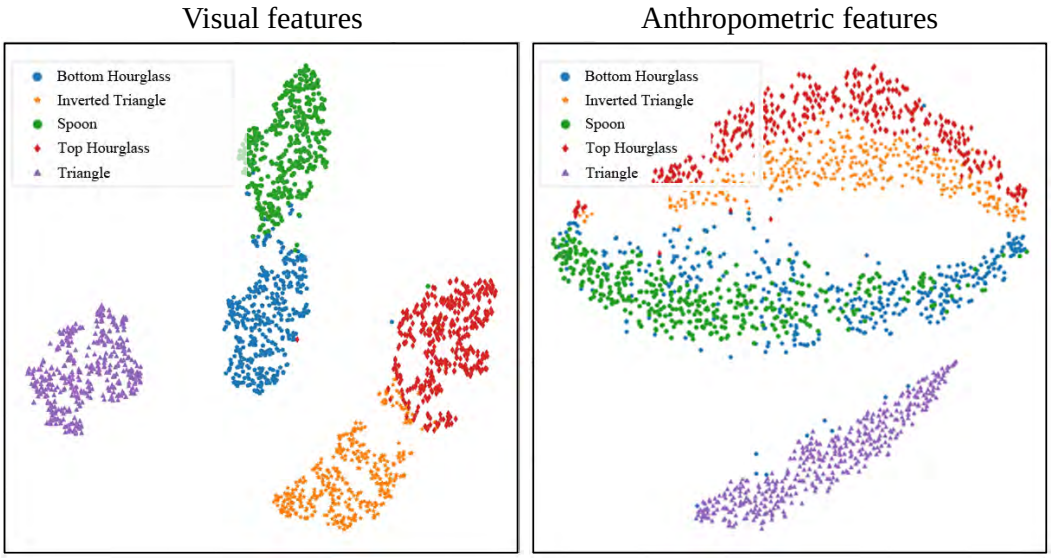


Figure 7.8 Visualization of different body features using t-SNE.

M-VTON system using the Type-aware dataset are visualized in Appendix B.

Comparing Visual and Anthropometric Features. Table 7.6 compares the performance of body shape classification, showing that the visual-based classification approach outperforms other baselines. This could be because other baselines use anthropometric data to classify body shapes while the proposed approach utilizes body images.

To further illustrate the difference between the visual and anthropometric features of the body shape, Figure 7.8 visualize their t-SNE [133] projection embeddings. The visual features are extracted from the frontal view images,

and the anthropometric features are measured from 3D models belonging to the testing set of the body shape dataset. It can be observed that the five body shapes are separated more clearly from each other in the left part of Figure 7.8 compared with anthropometric features in the right part. This suggests that the visual features contain more valuable information for characterizing the body shape. Moreover, the Euclidean distance between similar body shapes is closer. For instance, the distance between *inverted triangle* (orange star symbol) and *top hourglass* (red diamond symbol) is shorter than the distance between *inverted triangle* and *triangle* (purple triangle symbol). The main reason is that both *inverted triangle* and *top hourglass* body shapes have a wider upper body and a narrower lower body. In contrast, *triangle* body shape typically has larger hips. These results support the proposal that incorporating visual body features into learning body-shape-aware embeddings is effective.

7.6.4 Explainability Analysis (RQ3)

The attention maps of three query outfits are visualized in Figure 7.9. It provides a viewpoint of which fashion attributes the BA-Net focuses on when evaluating. Each row entry of the attention map represents attention weights $\boldsymbol{\alpha}_b$ generated in the JEM, which indicates the significance of fashion attributes with respect to corresponding body shapes. In the first two examples (Figures 7.9 (a) and (b)), two outfits are represented where the first query is incompatible with the *bottom hourglass, spoon,* and *triangle* body shapes. In contrast, the second query is compatible with them. The attention maps indicate that BA-Net attends mainly to the *bottom silhouette* attribute dimension (last row), *i.e.*, *Slim*, and *A-line*, respectively. This may be because these three body shapes all possess a larger hip measurement, congruent with an *A-line*

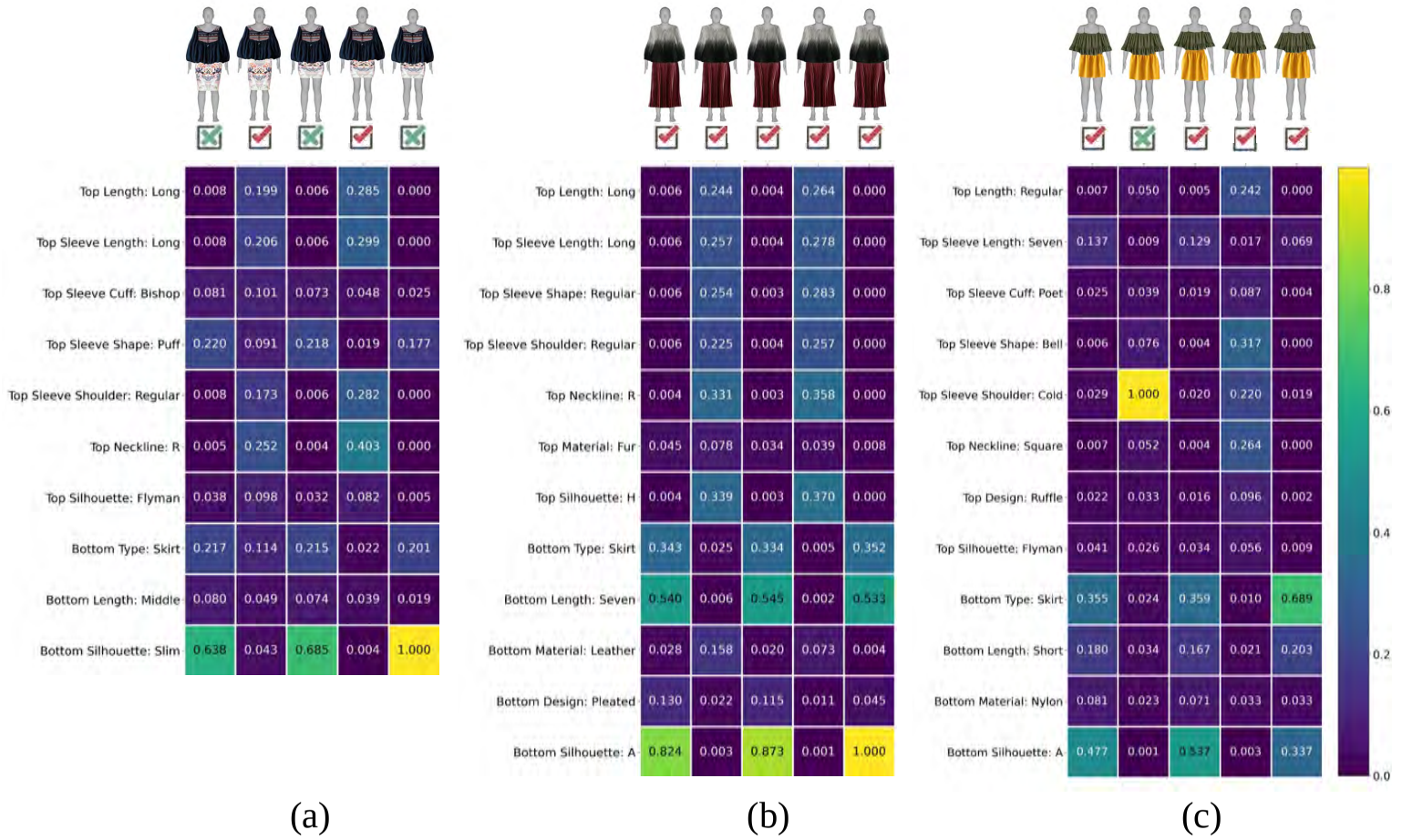


Figure 7.9 Visualization of attention maps computed in JEM. The vertical axis represents all the fashion attributes possessed by the query outfit. The horizontal axis represents five body shapes, namely, from left to right, *bottom hourglass, inverted triangle, spoon*, and *top hourglass*.

dress but not with a *slim* one. Additionally, Figure 7.9 (c) displays an outfit that is incompatible with *inverted triangle* body shape. BA-Net suggests that the main reason for this mismatch is that the top item contains a *cold shoulder* design. From a fashion perspective, this inference is reasonable because tops with *cold shoulder* designs often fail to provide adequate support for the chest and upper body, which can be a concern for individuals with a larger bust resulting in an unflattering and uncomfortable fit.

Interestingly, the BA-Net has varied focuses on fashion attributes belonging to the bottom and top items of different body shapes. The network concentrates mainly on the bottom attributes for body shapes such as *bottom hourglass, spoon*, and *triangle*. Conversely, it pays more attention to the top attributes for the *inverted triangle* and *top hourglass* body shapes. This could be because the bottom attributes play a more critical role in determining compatibility for body shapes with larger hip and thigh areas. On the other hand, for body shapes with broader shoulders and smaller waists, the network focuses more on the top attributes to ensure a balanced overall look that accentuates the waistline.

7.6.5 Perceptual Study (RQ4)

Finally, a perceptual study is conducted to show the potentiality of the BA-Net in practical applications. Specifically, ten experts working in the fashion industry are invited to assess the results of all the compatibility models from the following two aspects, (1) Body-shape-Aware Compatibility score (OCs): whether the outfits are compatible with the body shape or not; (2) Explanation Confidence score (ECs): whether the explanation reasonable or not. The score range is $[0, 1]$, 0.1 per level, and the final score is the weighted average of all

Table 7.7 Perceptual results of the compatibility models.

Methods	StyleMe [47]	TDRG [163]	M3TR [164]	CSRA [168]	FCN [98]	**BA-Net (Ours)**
OCs	49%	52%	51%	53%	59%	61%
ECs	-	-	-	-	-	67%

the scores given by those experts. The perceptual results are summarized in Table 7.7. It can be seen that the BA-Net enjoys the highest performance on Body-shape-Aware fashion compatibility while taking a unique advantage in explainability.

In addition to the perceptual study, the prototype for applying BA-Net in a real application is also built, as illustrated in Figure 7.10, showing the proposed method's practicality. Specifically, the prototype involves seven main steps for applying BA-Net in real applications:

(a). Inputting the personal information;

(b). Generating a 3D SMPL model according to the input measurements data;

(c). Adjusting and confirming the body shape;

(d). Browsing the fashion items;

(e). Selecting one favor clothing item with corresponding outfit recommendations that consider the body shape;

(f). Visualizing the outfit composition on the size of body shape;

(g). Translating the SMPL model into a human image via generative model such as Midjourney.

It can be seen that, with the awareness of body shape, customers can more easily and directly accept the recommended outfits. Furthermore, connecting

Figure 7.10 The pipeline of a prototype for applying BA-Net in a real application.

with the current cutting-edge techniques can generate more user-friendly and interesting results with substantial economic potential, e.g., translating the SMPL model into a human image via generative models such as Midjourney or executing a call API of Large Language models such as ChatGPT to make the explanation more like a natural conversation.

7.7 Chapter Summary

In conclusion, this chapter addresses the importance of considering body shape in outfit recommendations for real-life applications. The proposed BA-Net offers enhanced body-shape-aware embeddings to improve fashion cognition. A comprehensive dataset is constructed to provide diverse information about body shape. By incorporating visual features from body images, the body-shape embedding is strengthened. Within the BA-Net framework, the outfit is represented based on its try-on appearance, effectively capturing the scaling and spatial relationships between fashion items and the body. Experimental results on the O4U dataset validate the superior performance of BA-Net compared to existing state-of-the-art methods. Additionally, the ablation study validates the impact of the different component in the proposed approach. This chapter contributes to a more personalized and effective outfit recommendation system by considering the crucial aspect of body shape in the learning process

Chapter 8

Conclusions and Suggestions for Future Research

In this chapter, the conclusions of this thesis are first drawn, summarizing the key findings and contributions made throughout the research. Following the conclusions, the limitations of the research are discussed. Finally, the chapter ends with an outlook for future work.

8.1 Conclusions

The thesis has made progress in the field of fashion recommendation systems, addressing various aspects of this complex domain. In Chapter 3, the introduction of the A100 evaluation protocol provides valuable insights into the aesthetic ability of fashion compatibility models. By incorporating fine-grained indexes, A100 reveals specific areas where the models may need improvement, paving the way for future enhancements.

Chapter 4 focuses on the practical application of the fashion compatibility

model in real scenarios of online cross-selling. The proposed HON leveraging the multi-layer relations among fashion data achieves state-of-the-art performance compared to multiple baselines, offering a framework for integrating the trained fashion compatibility model into actual products for online cross-selling.

Chapter 5 presents a novel fashion compatibility model that combines the Bi-LSTM model and the inter-factor compatibility network. The system can accurately predict the convincing reason based on the outfit compatibility evaluation, contributing to the development of explainable fashion recommendation systems.

Chapter 6 introduces the task of Fashion Cognition Learning, aiming to learn the relationship between fashion outfits and personal physical information. A large-scale fashion outfit dataset is constructed for this task. The Fashion Convolutional Network is proposed to capture the relationships among visual-semantic embeddings and individuals' appearance features, enabling personalized fashion recommendations.

Chapter 7 addresses the importance of body shape in fashion recommendation by proposing BA-Net. The model learns better body-shape-aware embeddings by incorporating visual features extracted from body images and utilizing a comprehensive dataset on body shape. The outfit is encoded through its try-on appearance to enhance the model's performance, where the try-on images are generated through the proposed Multi-layer Virtual Try-on system.

8.2 Limitations

The first limitation is the reliance on currently available datasets, which may only partially encompass the wide range of fashion styles and individual prefer-

ences in real-world scenarios. Additionally, these datasets have certain biases or limitations in their coverage, potentially hindering the generalizability of the models to various fashion domains and diverse user populations. For example, the training data for the current models may not include sufficient representation of men's fashion, limiting the models' applicability in that domain. Therefore, it is crucial to consider the potential biases and limitations of the datasets when interpreting the findings and assessing the performance of the models, especially when applying them to contexts beyond the scope of the existing datasets.

The second limitation pertains to the Body-shape-Aware Network. BA-Net takes an approach that contradicts human habits to learn the correlation between fashion and body shape. In the real world, we typically assess the suitability of an outfit by physically trying it on and evaluating its compatibility with our specific body shape. However, BA-Net solely relies on evaluating compatibility based on 2D images. This disparity between BA-Net's evaluation approach and the natural evaluation process followed by humans introduces a potential limitation in terms of the model's generalizability and real-world applicability.

The third limitation pertains to the incomplete representation of physical attributes other than body shape, such as hairstyle, skin color, and other physical attributes. In this thesis, these attributes are predominantly described using textual information, which limits their effectiveness in capturing the visual aspects relevant to fashion compatibility. To address this limitation, expanding the research scope and developing more comprehensive representations encompassing these additional bodily features is crucial.

8.3 Suggestions for Future Research

Future work should address these limitations and explore new avenues in fashion recommendation systems.

Firstly, there is a need to expand the scope of available datasets to capture better the diversity of fashion styles and individual preferences, including underrepresented domains such as men's fashion. By incorporating a more comprehensive range of fashion attributes and user populations, the models can be more robust and applicable across various real-world fashion scenarios. Efforts should be made to collect more extensive and diverse datasets, ensuring that they reflect the richness and complexity of fashion styles.

Secondly, an area of future exploration involves the development of advanced virtual try-on systems that can automatically generate realistic outfit visualizations based on given clothing items and various body shapes, hairstyles, and other relevant attributes. By leveraging such advanced try-on techniques, one can simulate the appearance of individuals wearing different outfits, providing a more accurate representation of how clothes align with specific body characteristics. The compatibility between outfits and users can be evaluated by assessing the generated images of individuals wearing the outfits, allowing for a more objective and realistic application of personal stylists.

By addressing the limitations and pursuing these future directions, the personal stylist system can continue to evolve, providing more accurate, personalized, and immersive fashion recommendations that cater to diverse fashion domains, individual preferences, and real-world fashion scenarios.

References

[1] Kenan E Ak, Ashraf A Kassim, Joo Hwee Lim, and Jo Yew Tham. Learning attribute representations with localization for flexible fashion search. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7708–7717, 2018.

[2] Ziad Al-Halah, Rainer Stiefelhagen, and Kristen Grauman. Fashion forward: Forecasting visual style in fashion. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[3] Imran Amed, Johanna Andersson, Achim Berg, Martine Drageset, Saskia Hedrich, and Sara Kappelmark. The state of fashion 2018: Renewed optimism for the fashion industry, 2017.

[4] Imran Amed, Achim Berg, Anita Balchandani, Saskia Hedrich, Felix Rolkens, Robb Young, Jakob Ekelof Jensen, and Althea Peng. The state of fashion 2021, 2021.

[5] McKinsey BOF. The state of fashion report 2022. `https://www.businessoffashion.com/reports/news-analysis/the-state-of-fashion-2022-industry-report-bof-mckinsey/`, 2022. Accessed: (11 May 2023).

[6] Fangmei Chen and David Zhang. Combining a causal effect criterion for evaluation of facial attractiveness models. *Neurocomputing*, 177:98–109,

2016.

[7] I-Fei Chen and Chi-Jie Lu. Sales forecasting by combining clustering and machine-learning techniques for computer retailing. *Neural Computing and Applications*, 28:2633–2647, 2017.

[8] Wen Chen, Pipei Huang, Jiaming Xu, Xin Guo, Cheng Guo, Fei Sun, Chao Li, Andreas Pfadler, Huan Zhao, and Binqiang Zhao. Pog: personalized outfit generation for fashion recommendation at alibaba ifashion. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 2662–2670, 2019.

[9] Xu Chen, Hanxiong Chen, Hongteng Xu, Yongfeng Zhang, Yixin Cao, Zheng Qin, and Hongyuan Zha. Personalized fashion recommendation with visual explanations based on multimodal attention network: Towards visually explainable recommendation. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 765–774, 2019.

[10] Xu Chen, Yongfeng Zhang, Hongteng Xu, Yixin Cao, Zheng Qin, and Hongyuan Zha. Visually explainable recommendation. *preprint arXiv:1801.10288*, 2018.

[11] Zhao-Min Chen, Xiu-Shen Wei, Peng Wang, and Yanwen Guo. Multi-label image recognition with graph convolutional networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5177–5186, 2019.

[12] Zhongxia Chen, Xiting Wang, Xing Xie, Tong Wu, Guoqing Bu, Yining Wang, and Enhong Chen. Co-attentive multi-task learning for explainable recommendation. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, pages 2137–2143. AAAI Press,

2019.

[13] Wen-Huang Cheng, Sijie Song, Chieh-Yun Chen, Shintami Chusnul Hidayati, and Jiaying Liu. Fashion meets computer vision: A survey. *ACM Computing Surveys (CSUR)*, 54(4):1–41, 2021.

[14] Kat Collings. The foolproof way to find out your real body type. `http://www.whowhatwear.com/how-to-find-body-shape-calculator/`. Accessed: (12 Jan 2022).

[15] The Business Research Company. Ai in fashion global market report 2023. `https://www.researchandmarkets.com/reports/5767217/ai-in-fashion-global-market-report/`, 2023. Accessed: (15 May 2023).

[16] Charles Corbiere, Hedi Ben-Younes, Alexandre Ramé, and Charles Ollion. Leveraging weakly annotated data for fashion image retrieval and label prediction. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 2268–2274, 2017.

[17] Guillem Cucurull, Perouz Taslakian, and David Vazquez. Context-aware visual compatibility prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12617–12626, 2019.

[18] Cui et al. Dressing as a whole: Outfit compatibility learning based on node-wise graph neural networks. *arXiv*, 2019.

[19] Lavinia De Divitiis, Federico Becattini, Claudio Baecchi, and Alberto Del Bimbo. Disentangling features for fashion recommendation. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 2022.

[20] Lavinia De Divitiis, Federico Becattini, Claudio Baecchi, and Alberto

Del Bimbo. Disentangling features for fashion recommendation. *ACM Transactions on Multimedia Computing, Communications and Applications*, 19(1s):1–21, 2023.

[21] Ernani Viriato De Melo, Emilia Alves Nogueira, and Denise Guliato. Content-based filtering enhanced by human visual attention applied to clothing recommendation. In *2015 IEEE 27th International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 644–651. IEEE, 2015.

[22] Priya Devarajan and Cynthia L Istook. Validation of female figure identification technique (ffit) for apparel software. *Journal of Textile and Apparel, Technology and Management*, 4(1):1–23, 2004.

[23] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2625–2634, 2015.

[24] Xue Dong, Xuemeng Song, Fuli Feng, Peiguang Jing, Xin-Shun Xu, and Liqiang Nie. Personalized capsule wardrobe creation with garment and user modeling. In *Proceedings of the 27th ACM International Conference on Multimedia*, pages 302–310, 2019.

[25] Xue Dong, Jianlong Wu, Xuemeng Song, Hongjun Dai, and Liqiang Nie. Fashion compatibility modeling through a multi-modal try-on-guided scheme. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 771–780, 2020.

[26] Molly Eckman and Janet Wagner. Aesthetic aspects of the consumption of fashion design: The conceptual and empirical challenge. *ACR North*

American Advances, 1995.

[27] Molly Jean Eckman. *Consumers' aesthetic evaluation of clothing: The effect of age, sex, and fashion involvement*. PhD thesis, University of Maryland, College Park, 1992.

[28] Joanne Entwistle. *The aesthetic economy of fashion: Markets and value in clothing and modeling*. Berg, 2009.

[29] Zunlei Feng, Zhenyun Yu, Yezhou Yang, Yongcheng Jing, Junxiao Jiang, and Mingli Song. Interpretable partitioned embedding for customized multi-item fashion outfit composition. In *Proceedings of the 2018 ACM on International Conference on Multimedia Retrieval*, pages 143–151. ACM, 2018.

[30] Cherene Francis. Body shape calculator. `https://auraimageconsulting.com/body-shape-calculator/`. Accessed: (29 Dec 2022).

[31] Brendan J Frey and Delbert Dueck. Clustering by passing messages between data points. *science*, 315(5814):972–976, 2007.

[32] Bin-Bin Gao and Hong-Yu Zhou. Learning to discover multi-class attentional regions for multi-label image recognition. *IEEE Transactions on Image Processing*, 30:5920–5932, 2021.

[33] Lian Gao, Weixin Li, Zehua Huang, Di Huang, and Yunhong Wang. Automatic facial attractiveness prediction by deep multi-task learning. In *2018 24th International Conference on Pattern Recognition (ICPR)*, pages 3592–3597. IEEE, 2018.

[34] Ke Gong, Yiming Gao, Xiaodan Liang, Xiaohui Shen, Meng Wang, and Liang Lin. Graphonomy: Universal human parsing via graph transfer learning. In *Proceedings of the IEEE/CVF Conference on Computer*

Vision and Pattern Recognition, pages 7450–7459, 2019.

[35] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

[36] Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. Speech recognition with deep recurrent neural networks. In *2013 IEEE international conference on acoustics, speech and signal processing*, pages 6645–6649. IEEE, 2013.

[37] Alex Graves and Jürgen Schmidhuber. Framewise phoneme classification with bidirectional lstm and other neural network architectures. *Neural networks*, 18(5-6):602–610, 2005.

[38] Sida Gu, Xiaoqiang Liu, Lizhi Cai, and Jie Shen. Fashion coordinates recommendation based on user behavior and visual clothing style. In *Proceedings of the 3rd International Conference on Communication and Information Processing*, pages 185–189, 2017.

[39] Xiaoling Gu, Fei Gao, Min Tan, and Pai Peng. Fashion analysis and understanding with artificial intelligence. *Information Processing & Management*, 57(5):102276, 2020.

[40] Xintong Han, Zuxuan Wu, Weilin Huang, Matthew R Scott, and Larry S Davis. Finet: Compatible and diverse fashion image inpainting. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4481–4491, 2019.

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

[42] Xintong Han, Zuxuan Wu, Zhe Wu, Ruichi Yu, and Larry S Davis. Viton: An image-based virtual try-on network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7543–7552, 2018.

[43] Ruth Estella Hawthorne. *Aspects of design preference in clothing: aesthetic, motivation, and knowledge*. The Ohio State University, 1967.

[44] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[45] Ruining He and Julian McAuley. Vbpr: visual bayesian personalized ranking from implicit feedback. In *Proceedings of the AAAI conference on artificial intelligence*, volume 30, 2016.

[46] Sebastian Heinz, Christian Bracher, and Roland Vollgraf. An lstm-based dynamic customer model for fashion recommendation. *arXiv preprint arXiv:1708.07347*, 2017.

[47] Shintami Chusnul Hidayati, Ting Wei Goh, Ji-Sheng Gary Chan, Cheng-Chun Hsu, John See, Lai-Kuan Wong, Kai-Lung Hua, Yu Tsao, and Wen-Huang Cheng. Dress with style: Learning style from joint deep embedding of clothing styles and body shapes. *IEEE Transactions on Multimedia*, 23:365–377, 2020.

[48] Shintami Chusnul Hidayati, Cheng-Chun Hsu, Yu-Ting Chang, Kai-Lung Hua, Jianlong Fu, and Wen-Huang Cheng. What dress fits me best? fashion recommendation on the clothing style for personal body shape. In *Proceedings of the 26th ACM international conference on Multimedia*, pages 438–446, 2018.

[49] Min Hou, Le Wu, Enhong Chen, Zhi Li, Vincent W Zheng, and

Qi Liu. Explainable fashion recommendation: A semantic attribute region guided approach. *Twenty-Eight International Joint Conference on Artificial Intelligence*, 2019.

[50] Wei-Lin Hsiao and Kristen Grauman. Learning the latent "look": Unsupervised discovery of a style-coherent embedding from fashion images. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[51] Wei-Lin Hsiao and Kristen Grauman. Creating capsule wardrobes from fashion images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7161–7170, 2018.

[52] Wei-Lin Hsiao and Kristen Grauman. Vibe: Dressing for diverse body shapes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[53] Wei-Lin Hsiao, Isay Katsman, Chao-Yuan Wu, Devi Parikh, and Kristen Grauman. Fashion++: Minimal edits for outfit improvement. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5047–5056, 2019.

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

[55] Hyunwoo Hwangbo, Yang Sok Kim, and Kyung Jin Cha. Recommendation system development for fashion retail e-commerce. *Electronic Commerce Research and Applications*, 28:94–101, 2018.

[56] Tomoharu Iwata, Shinji Watanabe, and Hiroshi Sawada. Fashion coordinates recommender system using photographs from fashion magazines.

In *Twenty-Second International Joint Conference on Artificial Intelligence*, 2011.

[57] Vignesh Jagadeesh, Robinson Piramuthu, Anurag Bhardwaj, Wei Di, and Neel Sundaresan. Large scale visual recommendations from street fashion images. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1925–1934, 2014.

[58] Aline Jelenkovic, Reijo Sund, Yoon-Mi Hur, Yoshie Yokoyama, Jacob v B Hjelmborg, Sören Möller, Chika Honda, Patrik KE Magnusson, Nancy L Pedersen, Syuichi Ooki, et al. Genetic and environmental influences on height from infancy to early adulthood: An individual-based pooled analysis of 45 twin cohorts. *Scientific reports*, 6(1):28496, 2016.

[59] Jia Jia, Jie Huang, Guangyao Shen, Tao He, Zhiyuan Liu, Huanbo Luan, and Chao Yan. Learning to appreciate the aesthetic effects of clothing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30, 2016.

[60] Wang-Cheng Kang, Eric Kim, Jure Leskovec, Charles Rosenberg, and Julian McAuley. Complete the look: Scene-based complementary product recommendation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10532–10541, 2019.

[61] M Hadi Kiapour, Kota Yamaguchi, Alexander C Berg, and Tamara L Berg. Hipster wars: Discovering elements of fashion styles. In *European conference on computer vision*, pages 472–488. Springer, 2014.

[62] Donghyun Kim, Kuniaki Saito, Samarth Mishra, Stan Sclaroff, Kate Saenko, and Bryan A Plummer. Self-supervised visual attribute learning for fashion compatibility. In *Proceedings of the IEEE/CVF international*

conference on computer vision, pages 1057–1066, 2021.

[63] Seongjae Kim, Jinseok Seol, Holim Lim, and Sang-goo Lee. False negative distillation and contrastive learning for personalized outfit recommendation. *arXiv preprint arXiv:2110.06483*, 2021.

[64] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[65] Katrien Laenen, Susana Zoghbi, and Marie-Francine Moens. Web search of fashion items with multimodal querying. In *Proceedings of the eleventh ACM international conference on web search and data mining*, pages 342–350, 2018.

[66] Xingchen Li, Xiang Wang, Xiangnan He, Long Chen, Jun Xiao, and Tat-Seng Chua. Hierarchical fashion graph network for personalized outfit recommendation. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 159–168, 2020.

[67] Yang Li, Tong Chen, and Zi Huang. Attribute-aware explainable complementary clothing recommendation. *World Wide Web*, 24:1885–1901, 2021.

[68] Yuncheng Li, Liangliang Cao, Jiang Zhu, and Jiebo Luo. Mining fashion outfit composition using an end-to-end deep learning approach on set data. *IEEE Transactions on Multimedia*, 19(8):1946–1955, 2017.

[69] Zhi Li, Bo Wu, Qi Liu, Likang Wu, Hongke Zhao, and Tao Mei. Learning the compositional visual coherence for complementary recommendations. *arXiv preprint arXiv:2006.04380*, 2020.

[70] Lizi Liao, Xiangnan He, Bo Zhao, Chong-Wah Ngo, and Tat-Seng Chua. Interpretable multimodal retrieval for fashion products. In *Proceedings*

of the 26th ACM international conference on Multimedia, pages 1571–1579, 2018.

[71] Lin et al. Fashion outfit complementary item retrieval. In *CVPR*, 2020.

[72] Yujie Lin, Pengjie Ren, Zhumin Chen, Zhaochun Ren, Jun Ma, and Maarten de Rijke. Explainable fashion recommendation with joint outfit matching and comment generation. *arXiv preprint arXiv:1806.08977*, 2018.

[73] Yujie Lin, Pengjie Ren, Zhumin Chen, Zhaochun Ren, Jun Ma, and Maarten De Rijke. Explainable outfit recommendation with joint outfit matching and comment generation. *IEEE Transactions on Knowledge and Data Engineering*, 32(8):1502–1516, 2019.

[74] Zehang Lin, Haoran Xie, Peipei Kang, Zhenguo Yang, Wenyin Liu, and Qing Li. Cross-domain beauty item retrieval via unsupervised embedding learning. In *Proceedings of the 27th ACM International Conference on Multimedia*, pages 2543–2547, 2019.

[75] Jinhuan Liu, Xuemeng Song, Liqiang Nie, Tian Gan, and Jun Ma. An end-to-end attention-based neural model for complementary clothing matching. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 15(4):1–16, 2019.

[76] Si Liu, Jiashi Feng, Zheng Song, Tianzhu Zhang, Hanqing Lu, Changsheng Xu, and Shuicheng Yan. Hi, magic closet, tell me what to wear! In *Proceedings of the 20th ACM international conference on Multimedia*, pages 619–628, 2012.

[77] Xin Liu, Yongbin Sun, Ziwei Liu, and Dahua Lin. Learning diverse fashion collocation by neural graph filtering. *arXiv preprint arXiv:2003.04888*, 2020.

[78] Yining Liu and Yanming Shen. Personal tastes vs. fashion trends: predicting ratings based on visual appearances and reviews. *IEEE Access*, 6:16655–16664, 2018.

[79] Ziwei Liu, Ping Luo, Shi Qiu, Xiaogang Wang, and Xiaoou Tang. Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1096–1104, 2016.

[80] Ziwei Liu, Sijie Yan, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Fashion landmark detection in the wild. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*, pages 229–245. Springer, 2016.

[81] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, 34(6):248:1–248:16, October 2015.

[82] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. *ACM transactions on graphics (TOG)*, 34(6):1–16, 2015.

[83] Zhi Lu, Yang Hu, Yan Chen, and Bing Zeng. Personalized outfit recommendation with learnable anchors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12722–12731, 2021.

[84] Zhi Lu, Yang Hu, Yunchao Jiang, Yan Chen, and Bing Zeng. Learning binary code for personalized fashion recommendation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10562–10570, 2019.

[85] Minh-Thang Luong, Hieu Pham, and Christopher D Manning. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*, 2015.

[86] Utkarsh Mall, Kevin Matzen, Bharath Hariharan, Noah Snavely, and Kavita Bala. Geostyle: Discovering fashion trends and events. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 411–420, 2019.

[87] Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton Van Den Hengel. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 43–52. ACM, 2015.

[88] Wayne J Millar. Distribution of body weight and height: comparison of estimates based on self-reported and observed measures. *Journal of Epidemiology & Community Health*, 40(4):319–323, 1986.

[89] Samarth Mishra, Zhongping Zhang, Yuan Shen, Ranjitha Kumar, Venkatesh Saligrama, and Bryan A Plummer. Effectively leveraging attributes for visual similarity. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1015–1024, 2021.

[90] Dongmei Mo, Xingxing Zou, Kaicheng Pang, and Wai Keung Wong. Towards private stylists via personalized compatibility learning. *Expert Systems with Applications*, 219:119632, 2023.

[91] Dongmei Mo, Xingxing Zou, and WaiKeung Wong. Neural stylist: Towards online styling service. *Expert Systems with Applications*, 203:117333, 2022.

[92] Seyed Omid Mohammadi and Ahmad Kalhor. Smart fashion: A review

of ai applications in the fashion & apparel industry. *arXiv preprint arXiv:2111.00905*, 2021.

[93] Maryam Moosaei, Yusan Lin, and Hao Yang. Fashion recommendation and compatibility prediction using relational network. *arXiv preprint arXiv:2005.06584*, 2020.

[94] Takuma Nakamura and Ryosuke Goto. Outfit generation and style extraction via bidirectional lstm and autoencoder. *arXiv preprint arXiv:1807.03133*, 2018.

[95] Charles Packer, Julian McAuley, and Arnau Ramisa. Visually-aware personalized recommendation using interpretable image representations. *arXiv preprint arXiv:1806.09820*, 2018.

[96] Kaicheng Pang, Xingxing Zou, Fangjian Liao, and Waikeung Wong. M-vton: Multi-layer virtual try-on system. *Design and Semantics of Form and Movement*, 266, 2023.

[97] Kaicheng Pang, Xingxing Zou, and Waikeung Wong. Modeling fashion compatibility with explanation by using bidirectional lstm. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 3894–3898, June 2021.

[98] Kaicheng Pang, Xingxing Zou, and Waikeung Wong. Dress well via fashion cognitive learning. In *33rd British Machine Vision Conference 2022, BMVC 2022, London, UK, November 21-24, 2022*, page 251. BMVA Press, 2022.

[99] Kaicheng Pang, Xingxing Zou, and Waikeung Wong. Towards intelligent online cross-selling. *Expert Systems with Applications*, 2022.

[100] Kaicheng Pang, Xingxing Zou, and Waikeung Wong. Learning visual body-shape-aware embeddings for fashion compatibility. In *Proceedings*

of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), 2024.

[101] Christopher J Parker, Steven George Hayes, Kathryn Brownbridge, and Simeon Gill. Assessing the female figure identification technique's reliability as a body shape classification system. *Ergonomics*, 64(8):1035–1051, 2021.

[102] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[103] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[104] Bryan A Plummer, Mariya I Vasileva, Vitali Petsiuk, Kate Saenko, and David Forsyth. Why do these match? explaining the behavior of image similarity models. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16*, pages 652–669. Springer, 2020.

[105] Luisa F Polanía and Satyajit Gupte. Learning fashion compatibility across apparel categories for outfit recommendation. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 4489–4493. IEEE, 2019.

[106] Albert Pumarola, Antonio Agudo, Alberto Sanfeliu, and Francesc Moreno-Noguer. Unsupervised person image synthesis in arbitrary poses.

In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8620–8628, 2018.

[107] Yurui Ren, Xiaoming Yu, Junming Chen, Thomas H Li, and Ge Li. Deep image spatial transformation for person image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7690–7699, 2020.

[108] Herbert E. Robbins. A stochastic approximation method. *Annals of Mathematical Statistics*, 22:400–407, 2007.

[109] Tao Ruan, Ting Liu, Zilong Huang, Yunchao Wei, Shikui Wei, and Yao Zhao. Devil in the details: Towards accurate single and multiple human parsing. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 4814–4821, 2019.

[110] Sebastian Ruder. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*, 2016.

[111] Dikshant Sagar, Jatin Garg, Prarthana Kansal, Sejal Bhalla, Rajiv Ratn Shah, and Yi Yu. Pai-bpr: Personalized outfit recommendation scheme with attribute-wise interpretability. In *2020 IEEE Sixth International Conference on Multimedia Big Data (BigMM)*, pages 221–230. IEEE, 2020.

[112] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 618–626, 2017.

[113] Akash Sengupta. 3d body measurements.

[114] Edward Shen, Henry Lieberman, and Francis Lam. What am i gonna

wear? scenario-oriented recommendation. In *Proceedings of the 12th international conference on Intelligent user interfaces*, pages 365–368, 2007.

[115] Shengjie Shi, Fei Gao, Xuantong Meng, Xingxin Xu, and Jingjie Zhu. Improving facial attractiveness prediction via co-attention learning. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4045–4049. IEEE, 2019.

[116] Yong-Siang Shih, Kai-Yueh Chang, Hsuan-Tien Lin, and Min Sun. Compatibility family learning for item recommendation and generation. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[117] Aliaksandr Siarohin, Enver Sangineto, Stéphane Lathuiliere, and Nicu Sebe. Deformable gans for pose-based human image generation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3408–3416, 2018.

[118] Karla Kristin Peavy Simmons. *Body shape analysis using three-dimensional body scanning technology*. North Carolina State University, 2002.

[119] Edgar Simo-Serra, Sanja Fidler, Francesc Moreno-Noguer, and Raquel Urtasun. Neuroaesthetics in fashion: Modeling the perception of fashionability. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 869–877, 2015.

[120] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[121] Xuemeng Song, Fuli Feng, Jinhuan Liu, Zekun Li, Liqiang Nie, and Jun Ma. Neurostylist: Neural compatibility modeling for clothing matching.

In *Proceedings of the 2017 ACM on Multimedia Conference*, pages 753–761. ACM, 2017.

[122] Xuemeng Song, Xianjing Han, Yunkai Li, Jingyuan Chen, Xin-Shun Xu, and Liqiang Nie. Gp-bpr: Personalized compatibility modeling for clothing matching. In *Proceedings of the 27th ACM International Conference on Multimedia*, pages 320–328, 2019.

[123] Omprakash Sonie, Muthusamy Chelliah, and Shamik Sural. Personalised fashion recommendation using deep learning. In *Proceedings of the ACM India Joint International Conference on Data Science and Management of Data*, pages 368–368, 2019.

[124] Tianyu Su, Xuemeng Song, Na Zheng, Weili Guan, Yan Li, and Liqiang Nie. Complementary factorization towards outfit compatibility modeling. In *Proceedings of the 29th ACM international conference on multimedia*, pages 4073–4081, 2021.

[125] Guang-Lu Sun, Xiao Wu, and Qiang Peng. Part-based clothing image annotation by visual neighbor retrieval. *Neurocomputing*, 213:115–124, 2016.

[126] Jie Sun, Qianyun Cai, Tao Li, Lei Du, and Fengyuan Zou. Body shape classification and block optimization based on space vector length. *International Journal of Clothing Science and Technology*, 31(1):115–129, 2019.

[127] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. *Advances in neural information processing systems*, 27, 2014.

[128] Reuben Tan, Mariya I Vasileva, Kate Saenko, and Bryan A Plummer. Learning similarity conditions without explicit supervision. In *Proceed-*

ings of the IEEE/CVF International Conference on Computer Vision, pages 10373–10382, 2019.

[129] Pongsate Tangseng and Takayuki Okatani. Toward explainable fashion recommendation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2153–2162, 2020.

[130] Pongsate Tangseng and Takayuki Okatani. Toward explainable fashion recommendation. In *The IEEE Winter Conference on Applications of Computer Vision (WACV)*, March 2020.

[131] Pongsate Tangseng, Kota Yamaguchi, and Takayuki Okatani. Recommending outfits from personal closet. In *2017 IEEE International Conference on Computer Vision Workshop (ICCVW)*, pages 2275–2279. IEEE, 2017.

[132] Adane Nega Tarekegn, Mario Giacobini, and Krzysztof Michalak. A review of methods for imbalanced multi-label classification. *Pattern Recognition*, 118:107965, 2021.

[133] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

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

[135] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[136] Andreas Veit, Serge Belongie, and Theofanis Karaletsos. Conditional

similarity networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 830–838, 2017.

[137] Andreas Veit, Balazs Kovacs, Sean Bell, Julian McAuley, Kavita Bala, and Serge Belongie. Learning visual clothing style with heterogeneous dyadic co-occurrences. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4642–4650, 2015.

[138] S. Vittayakorn, A. C. Berg, and T. L. Berg. When was that made? In *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 715–724, Los Alamitos, CA, USA, mar 2017. IEEE Computer Society.

[139] Jiang Wang, Yi Yang, Junhua Mao, Zhiheng Huang, Chang Huang, and Wei Xu. Cnn-rnn: A unified framework for multi-label image classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2285–2294, 2016.

[140] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, et al. Deep high-resolution representation learning for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 43(10):3349–3364, 2020.

[141] Wenguan Wang, Zhijie Zhang, Siyuan Qi, Jianbing Shen, Yanwei Pang, and Ling Shao. Learning compositional neural information fusion for human parsing. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5703–5713, 2019.

[142] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7794–7803, 2018.

[143] Xin Wang, Bo Wu, Yun Ye, and Yueqi Zhong. Outfit compatibility prediction and diagnosis with multi-layered comparison network. In *Proceedings of the 2019 ACM on Multimedia Conference*. ACM, 2019.

[144] Yufan Wen, Xiaoqiang Liu, and Bo Xu. Personalized clothing recommendation based on knowledge graph. In *2018 International Conference on Audio, Language and Image Processing (ICALIP)*, pages 1–5. IEEE, 2018.

[145] Agung Toto Wibowo, Advaith Siddharthan, Judith Masthoff, and Chenghua Lin. Incorporating constraints into matrix factorization for clothes package recommendation. In *Proceedings of the 26th Conference on User Modeling, Adaptation and Personalization*, pages 111–119, 2018.

[146] Qianqian Wu, Pengpeng Zhao, and Zhiming Cui. Visual and textual jointly enhanced interpretable fashion recommendation. *IEEE Access*, 8:68736–68746, 2020.

[147] Lumin Xu, Yingda Guan, Sheng Jin, Wentao Liu, Chen Qian, Ping Luo, Wanli Ouyang, and Xiaogang Wang. Vipnas: Efficient video pose estimation via neural architecture search. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16072–16081, 2021.

[148] Sijie Yan, Ziwei Liu, Ping Luo, Shi Qiu, Xiaogang Wang, and Xiaoou Tang. Unconstrained fashion landmark detection via hierarchical recurrent transformer networks. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 172–180, 2017.

[149] Chaojie Yang, Hanhui Li, Shengjie Wu, Shengkai Zhang, Haonan Yan, Nianhong Jiao, Jie Tang, Runnan Zhou, Xiaodan Liang, and Tianxiang Zheng. Bodygan: General-purpose controllable neural human body

generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7733–7742, 2022.

[150] Xin Yang, Xuemeng Song, Fuli Feng, Haokun Wen, Ling-Yu Duan, and Liqiang Nie. Attribute-wise explainable fashion compatibility modeling. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 17(1):1–21, 2021.

[151] Xuewen Yang, Dongliang Xie, Xin Wang, Jiangbo Yuan, Wanying Ding, and Pengyun Yan. Learning tuple compatibility for conditional outfit recommendation. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 2636–2644, 2020.

[152] Xun Yang, Xiangnan He, Xiang Wang, Yunshan Ma, Fuli Feng, Meng Wang, and Tat-Seng Chua. Interpretable fashion matching with rich attributes. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 775–784, 2019.

[153] Xun Yang, Yunshan Ma, Lizi Liao, Meng Wang, and Tat-Seng Chua. Transnfcm: Translation-based neural fashion compatibility modeling. *arXiv preprint arXiv:1812.10021*, 2018.

[154] Xun Yang, Yunshan Ma, Lizi Liao, Meng Wang, and Tat-Seng Chua. Transnfcm: Translation-based neural fashion compatibility modeling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 403–410, 2019.

[155] Jeong Yim Lee, Cynthia L Istook, Yun Ja Nam, and Sun Mi Park. Comparison of body shape between usa and korean women. *International Journal of Clothing Science and Technology*, 19(5):374–391, 2007.

[156] Cong Yu, Yang Hu, Yan Chen, and Bing Zeng. Personalized fashion

design. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.

[157] Yuhui Yuan, Xiaokang Chen, Xilin Chen, and Jingdong Wang. Segmentation transformer: Object-contextual representations for semantic segmentation. *arXiv preprint arXiv:1909.11065*, 2019.

[158] Yuhui Yuan, Xilin Chen, and Jingdong Wang. Object-contextual representations for semantic segmentation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VI 16*, pages 173–190. Springer, 2020.

[159] Joe Yue-Hei Ng, Matthew Hausknecht, Sudheendra Vijayanarasimhan, Oriol Vinyals, Rajat Monga, and George Toderici. Beyond short snippets: Deep networks for video classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4694–4702, 2015.

[160] Huijing Zhan, Jie Lin, Kenan Emir Ak, Boxin Shi, Ling-Yu Duan, and Alex C Kot. A3-fkg: Attentive attribute-aware fashion knowledge graph for outfit preference prediction. *IEEE Transactions on Multimedia*, 2021.

[161] Heming Zhang, Xuewen Yang, Jianchao Tan, Chi-Hao Wu, Jue Wang, and C-C Jay Kuo. Learning color compatibility in fashion outfits. *arXiv preprint arXiv:2007.02388*, 2020.

[162] Xishan Zhang, Jia Jia, Ke Gao, Yongdong Zhang, Dongming Zhang, Jintao Li, and Qi Tian. Trip outfits advisor: Location-oriented clothing recommendation. *IEEE Transactions on Multimedia*, 19(11):2533–2544, 2017.

[163] Jiawei Zhao, Ke Yan, Yifan Zhao, Xiaowei Guo, Feiyue Huang, and Jia Li. Transformer-based dual relation graph for multi-label image recog-

nition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 163–172, 2021.

[164] Jiawei Zhao, Yifan Zhao, and Jia Li. M3tr: Multi-modal multi-label recognition with transformer. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 469–477, 2021.

[165] Haitian Zheng, Kefei Wu, Jong-Hwi Park, Wei Zhu, and Jiebo Luo. Personalized fashion recommendation from personal social media data: An item-to-set metric learning approach. In *2021 IEEE International Conference on Big Data (Big Data)*, pages 5014–5023. IEEE, 2021.

[166] Na Zheng, Xuemeng Song, Qingying Niu, Xue Dong, Yibing Zhan, and Liqiang Nie. Collocation and try-on network: Whether an outfit is compatible. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 309–317, 2021.

[167] Dongliang Zhou, Haijun Zhang, Kai Yang, Linlin Liu, Han Yan, Xiaofei Xu, Zhao Zhang, and Shuicheng Yan. Learning to synthesize compatible fashion items using semantic alignment and collocation classification: An outfit generation framework. *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

[168] Ke Zhu and Jianxin Wu. Residual attention: A simple but effective method for multi-label recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 184–193, 2021.

[169] Xingxing Zou, Zhizhong Li, Ke Bai, Dahua Lin, and Waikeung Wong. Regularizing reasons for outfit evaluation with gradient penalty. *arXiv preprint arXiv:2002.00460*, 2020.

[170] Xingxing Zou, Kaicheng Pang, Wen Zhang, and Waikeung Wong. How good is aesthetic ability of a fashion model? In *Proceedings of the*

IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 21200–21209, 2022.

[171] Xingxing Zou, Wai Keung Wong, Can Gao, and Jie Zhou. Foco system: a tool to bridge the domain gap between fashion and artificial intelligence. *International Journal of Clothing Science and Technology*, 2019.

[172] Xingxing Zou and Waikeung Wong. fashion after fashion: A report of ai in fashion. *arXiv preprint arXiv:2105.03050*, 2021.

Appendix A

Limitations of Existing FITB Tests

This section primarily serves as a supplement to Chapter 3.2 in the main thesis, providing visual examples to further illustrate the limitations of existing FITB (Fill-in-the-Blank) tests in evaluating aesthetic ability.

The FITB test was first employed as the fashion compatibility modeling task metric in [41]. Subsequently, it quickly gained popularity and became a mainstream evaluation approach in this field. Figure A.1 randomly showcases example questions from the Maryland FITB test. Several observations can be made:

1. The FITB test contains some unrelated images, making it less clean. For instance, in the first question, the items depicted belong to furniture rather than fashion items.

2. Incorrect answers are easily eliminated from the choice set based on the principle of creating a complete outfit. In the second example in Figure A.1, the "black hat" cannot form a complete outfit when combined

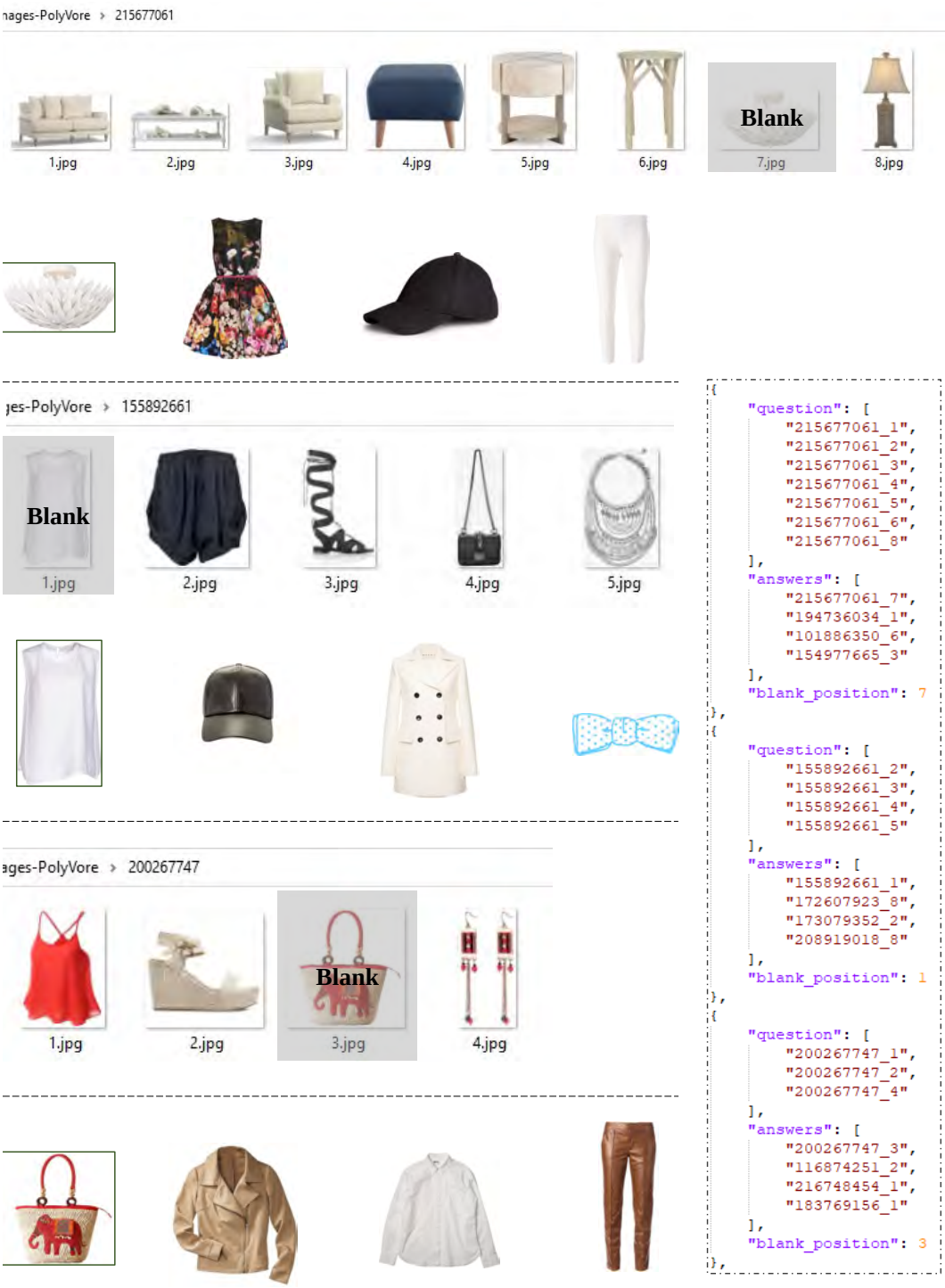


Figure A.1 Some FITB questions contained in the Maryland testing set [41]. The correct answer is indicated by a green box.

with the other items in the question.

3. The original outfit used to generate a particular question may not be valid. As demonstrated in the third example in Figure A.1, it is evident that the bottom part is missing. If factors related to aesthetic considerations are excluded, the correct choice among the four candidates should be the "brown pants" instead of the bag.

Vasileva *et al.* [134] addressed the aforementioned issues by introducing the Type-aware dataset, which includes fine-grained item types. In contrast to the Maryland FITB test with 3,076 questions, the Type-aware FITB test comprises a larger set of 10,000 questions. Furthermore, the creation of incorrect choices in each Type-aware test question differs from the previous approach. In this case, the incorrect choices are sampled from items within the same category as the correct choice. Server examples in the Type-aware FITB testing set are visualized in Figure A.2, which reveals several limitations:

1. The examined aspects of each option are not uniform. Taking the first question in Figure A.2 as an example, the second choice can be excluded due to its incompatible **color** with the question. Similarly, the **print** of both the first and third choices is incompatible.

2. The randomly generated chosen set seems questionable. As demonstrated in the second and third cases in Figure A.2, there could be an alternative option. For instance, the "black bag" could also be considered compatible, as its silhouette has a style more similar to the question than the pink flap bag.

3. The Type-aware dataset still contains unrelated images and invalid questions. The dataset initially consists of 68,306 outfits and 365,054 items.

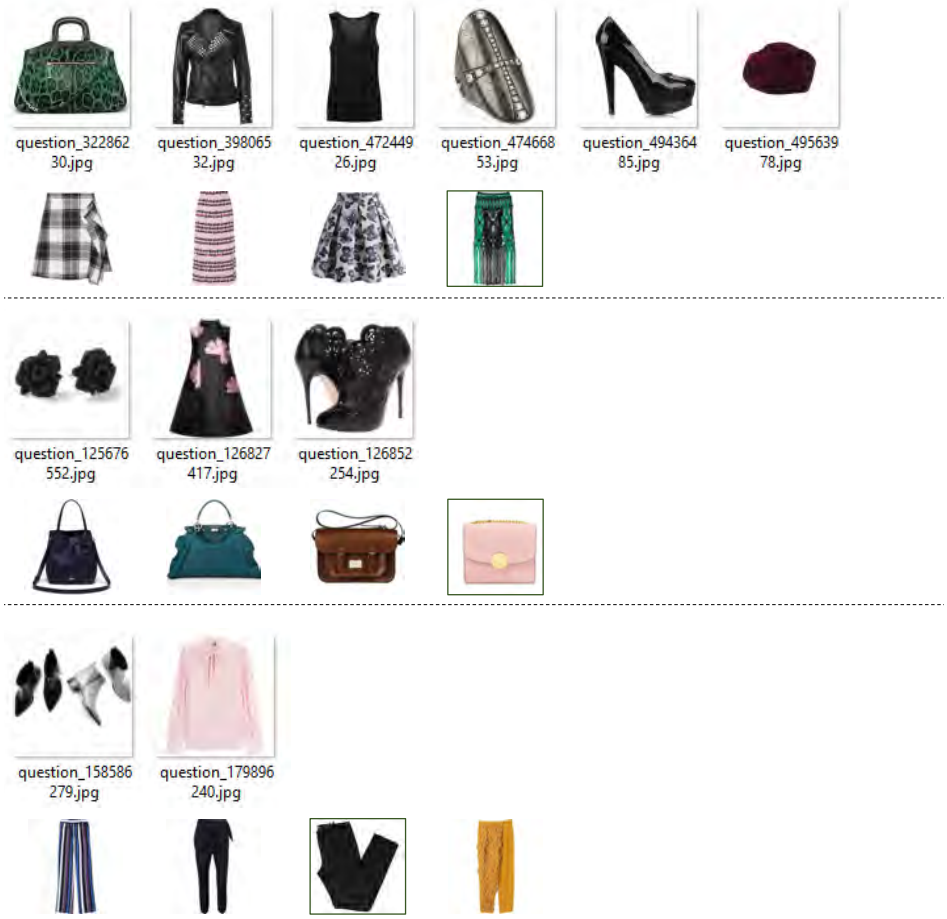


Figure A.2 FITB questions contained in the Type-aware testing set [134]. The correct answer is indicated by a green box.

After the cleaning process, the remaining number of items is 206,656. Additionally, the Type-aware FITB test involves 10,000 outfits, of which 1,875 are deemed invalid.

In addition to the mainstream FITB tests mentioned earlier, other FITB tests are introduced in various research papers. These tests are derived from newly introduced outfit datasets. For example, iFashion [8], a dataset collected from Taobao.com, includes the iFashion FITB test, which follows a strategy

similar to the Maryland test. In this case, 10% of the data is designated as the test set. For each masked item, three items are randomly selected from other outfits, along with the ground truth item, to create a multiple-choice set. The FashionVC test [121], on the other hand, only includes top and bottom images, while the Polyvore-U dataset consists of top, bottom, and shoe images. Upon careful investigation, the following conclusions can be drawn: 1). The aesthetic standard in all existing FITB tests is highly diverse and lacks collective consensus. This is attributed to the fact that the outfits used to create the questions are generated by different online users. 2). The method used to create the choice set raises concerns. There is a significant possibility that the masked item may not be the most compatible option among the available choices. 3). None of these FITB tests systematically reflect the fashion aesthetic standard.

Appendix B

Qualitative Results of M-VTON on Type-aware Dataset

This section serves as a supplement to Chapter 7 in the main thesis, providing qualitative results of evaluating M-VTON on the Type-aware dataset [134]. The type-aware Polyvore dataset contains 32,140 outfits and 175,485 item images. Unlike the O4U dataset, outfits in the type-aware dataset have a varied number of items, and there are 11 fashion categories, including outerwear, all-body, and top. Therefore, this subsection shows the try-on images with different item combinations. Figure B.1 (a) shows the results of trying on a dress, and Figure B.1 (b) presents the try-on images of outfits containing one top and one bottom. Figure B.2 (c) shows more interesting results since these dresses should cover the outerwear. Figure B.2 (d) shows the most complicated cases. Outfits are comprised of three garments, and these try-on results show that the proposed M-VTON can accurately demonstrate multi-layers of an outfit in a predefined order.

(a). Outfit containing dress only

(b). Outfit containing two items

Figure B.1 Try-on results of outfits from the Type-aware Polyvore [134] dataset. (a). Each outfit contains one item. (b). Each outfit contains two items.

(c). Outfit containing outerwear and dress

(d). Outfit containing three items

Figure B.2 Try-on results of outfits from the Type-aware Polyvore [134] dataset. (c). Each outfit contains outerwear and a dress. (d). Each outfit contains three items.