Using Bi-Directional Instance-Based Compatibility Prediction for Outfit Recommendation

Tzung-Pei Hong
Department of Computer Science and Information Engineering,
National University of Kaohsiung, Taiwan
Department of Computer Science and Engineering,
National Sun Yat-sen University, Taiwan
Email: tphong@nuk.edu.tw

Yun-Pei Chao
Department of Computer Science and Engineering,
National Sun Yat-sen University, Taiwan
Email: chrischao109@gmail.com

Jiann-Shu Lee
Department of Computer Science and Information Engineering,
National University of Tainan, Taiwan
Email: cslee@mail.nutn.edu.tw

Ja-Hwung Su*
Department of Computer Science and Information Engineering,
National University of Kaohsiung, Taiwan
**Email: bb0820@ms22.hinet.net*

***Abstract*—Existing fashion recommendation studies focus primarily on recommending individual items. However, this paradigm cannot cater to user needs on fashionable outfit. To obtain a fashionable and well-coordinated outfit, outfit recommendation focuses not only on one item but on all items in an outfit. Such fashion recommendation outputs multiple images of items to constitute a whole outfit. To this end, this paper proposes a novel outfit recommendation method named Bi-directional Instance-based Compatibility Prediction (BICP) suggesting suitable revised outfits based on the outfit inputs of users. In this method, the conditional Bi-directional Long Short-Term Memory (Bi-LSTM) mechanism is used as a backbone to generate the embedding representation of fashion items. To approximate the best outfit, a new metric called I2I-cos (Instance-to-Instance) cosine similarity is also proposed for outfit compatibility calculation. Finally, we made distribution diagrams indicating the outfits recommended by the proposed approaches better align with people's aesthetics and preferences.**

Keywords- outfit recommendation; fashion compatibility; Bi-LSTM; deep learning.

I. INTRODUCTION

Fashion is a form of self-expression and autonomy that dictates what we wear, including clothing, footwear, bags, and accessories. With the rise of fashion e-commerce, people can sell and buy apparel online. Therefore, online retailers have invested significant resources to implement machine learning techniques for fashion recommendation. Existing studies for fashion recommendation can be broadly split into two groups: complementary item recommendation [9][10][11] and outfit recommendation [1]. Generally, complementary item recommendation is proposed to suggest a single item for some things that have been matched. Outfit recommendation actually recommends a full set of coordinated items to form an outfit. The most recent studies primarily focus on complementary item recommendation, overlooking outfit recommendation.

However, outfit recommendation must take into account the compatibility of items to suggest suitable outfits. Therefore, modeling the compatibility of items is the key to outfit recommendation. Figure 1 illustrates the examples of compatible and incompatible outfits. In this paper, we propose a novel recommendation mechanism for outfit recommendation to suggest suitable revised outfits corresponding to the given category based on the outfit inputs of users. In addition, we also propose a new metric for outfit compatibility prediction in these recommended outfits. Finally, we conducted an objective evaluation of the recommended outfits through distribution diagrams to understand whether the outfits recommended by our methods align with human aesthetics.

Fig. 1. Examples of (a): compatible outfits and (b): incompatible outfits.

The remainder of this paper is structured in the following. The related research is briefly reviewed in Section 2. In Section 3, the proposed method for outfit recommendation and the metric for outfit compatibility prediction are presented in detail. The experimental analysis is interpreted in Section 4. Finally, the conclusions and future works are shown in Section 5.

II. RELATED WORK

As shown in Figure 2, research on Fashion Compatibility Modeling (FCM) can be roughly categorized into pairwise-based [6], sequence-based [5], and graph-based [3] methods. Pairwise-based methods focus primarily on the compatibility between two given items. For example, Song et al. [12] proposed a multi-modal pairwise compatibility modeling scheme with a dual auto-encoder network to match the top and bottom of the outfit. Sequence-based methods think of an outfit as a sequence or a set and each item in the outfit as

 ISBN: 978-1-68558-182-4

a time step and model the task as a sequence problem to uncover complex compatibility relationships among items. Han et al. [8] proposed sequentially modeling the compatibility of items in a given outfit with a Bi-LSTM model to carry out a fashion compatibility prediction task, mainly performing two tasks: complementary item recommendation and compatibility prediction. This study regarded an outfit as a specific ordered sequence of fashion items' images. For a sequence of images, the goal is to recommend suitable items in any position of sequence. Bi-LSTM [4] is proposed for Natural Language Processing, containing forward and backward LSTMs. In the forward direction, Bi-LSTM predicts the feature distribution of the next item based on the previous image features, and Bi-LSTM predicts the feature distribution of the previous item based on the image features in the backward direction. Graph-based methods have recently attracted attention as they excel at enhanced item relations. Such methods model the outfit as a graph in which nodes represent outfit items and node edges represent relations between items. Given this graph, graph neural networks are used to calculate outfit compatibility. Cucurull et al. [2] utilized a graph neural network to learn item embeddings conditioned on their context and cast the FCM task as an edge prediction problem. Iyer et al. [7] embedded a bi-level graph attention mechanism into a graph neural network, increasing the prediction quality. Wang et al. [14] aimed at the heterogeneous graph neural network using the hierarchical attention mechanism.

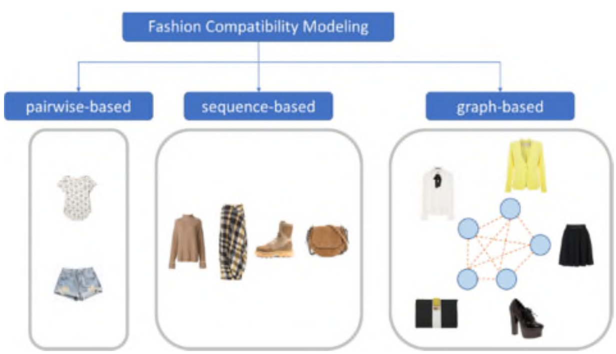

Fig. 2. Three kinds of fashion recommendation.

III. PROPOSED METHOD

In this section, we will present the details of proposed method, including preliminary definition, framework and compatibility calculation.

A. Preliminary

In the proposed method, we employ the pre-trained Bi-LSTM model as the expert-like embedded model which is extended from Han et al.'s approach. We also employ Han et al.'s compatibility prediction concept [8] to propose two novel compatibility methods to evaluate outfit compatibility, which focus on the compatibility effect of each outfit. The major uniqueness of this paper include: 1) the first and the last fashion items are fixed, and 2) our approaches use faster compatibility calculation methods than Han et al.'s. Here, we define an outfit O as a sequence $(I_1, I_2, \ldots, I_N)$, where I_j is the j-th fashion item, and N is the number of items in the outfit. We adopt the pre-processing method for a set of items' images proposed by Han et al. using the pre-trained Inception-V3 model [13] on ImageNet to extract their feature vectors. Thus, we re-define an outfit $X = (X_1, X_2, \ldots, X_N)$ where X_j is the feature-vector representation of the j-th fashion item in the outfit. Note that O and X have variable lengths because different outfits may have different numbers of items.

B. Overview

The method framework is shown in Figure 3. It performs compatibility prediction based on the Bi-LSTM framework during inference time. An outfit formed by the item images is treated as a sequence, and the images are extracted by the pre-trained Inception-V3 model on ImageNet separately and then input into the pre-trained Bi-LSTM model to sequentially predict the next item conditioned on previously seen items — both forward and backward — calculating the similarity of the features to accomplish outfit compatibility prediction. Next, the BICP approach is executed. In BICP, we fix an outfit's head and tail items in the prediction process, meaning the first and last items of the new outfit are the same as those in the original input outfit. The middle part of a new outfit is formed by combining the predictions from the two expert models. Then, the Instance-to-Instance (I2I) similarity indicating the cosine similarity between two instances is calculated. Finally, the suitable outfit and its compatibility score are returned.

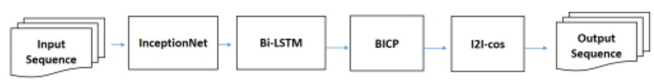

Fig. 3. Framework of the proposed method.

C. Bi-directional Instance-based Compatibility Prediction (BICP)

We use the Bi-LSTM characteristics to form new outfits. Since Bi-LSTM predicts feature vectors in the next positions in a bidirectional manner, the head and tail items are two main constraints to initialize the process. Therefore, in the proposed approach, we fix an outfit's head and tail items in the prediction process, meaning the first and last items of the new outfit are the same as those in the original input outfit. The middle part of a new outfit is formed by combining the predictions from the two expert models, similar to Han et al.'s complementary item recommendation approach [8]. Yet, we directly calculate feature similarity scores using cosine similarity to expedite the recommendation process. In addition, we restrict the recommended items to be of the same category as the items in the input outfit.

Therefore, the middle part of the new outfit is formed in the following way. For generating the item at the t-th position in an output outfit, our method uses the Bi-LSTM model to predict the FW feature vector and the BW feature vector of the item. In the forward direction, given the first t-1 items, X_1 to $X_{t\text{-}1}$, FW predicts the feature vector $H_{t\text{-}1}$ of the item X_t at the t-th position. In the backward direction, given the items of X_N to X_{t+1}, BW predicts the feature vector $\widetilde{H}_{t+1}$. Formally, the item' feature vector in the t-th position (except the head and the tail) of the output outfit is built as follows:

$$X'_t = \arg\max_{Y_k \in C_t} \left(FWScore(H_{t-1}, Y_k) + BWScore(\widetilde{H}_{t+1}, Y_k)\right), \quad (1)$$

where t is the position that we seek to adopt as a sequential instance, which is between 2 to N-1. C_t dataset (choice set) is formed by the same category as the item X_t at the t-th position, and each item Y_k is processed to extract feature vectors using the pre-trained Inception-V3. We use the cosine measure (*cos*) to calculate the similarity between H_{t-1} and Y_k to calculate the FW feature score (denoted by *FWScore*). We also do the same for $\widetilde{H}_{t+1}$ to get a BW feature score (denoted by *BWScore*). Hence, FW and BW expert models independently calculate the similarity of one candidate belonging to the outfit, and the candidate with the highest total score is selected at the t-th position. Now, X'_t represents the new item's feature vector. We obtain the feature vector of the item and retrieve the original image of this item I'_t. We thus form the middle part of a new image-form outfit.

D. *Compatibility Calculation: Instance-to-Instance cos (I2I-cos)*

After performing BICP, we obtain a new image-form outfit. Ideally, the newly generated outfits by the model are the same as the input outfit, indicating that the model considers this outfit to be the most suitable combination. We also employ the concept of Han et al.'s compatibility prediction to assess the overall outfit compatibility by computing feature similarities.

Algorithm 1:
The Procedure of BICP Evaluated using the I2I-cos Method

Input: A sequence of outfit items O and its outfit length N
Output: A suitable outfit and its compatibility score

```
new_outfit O' = [];
X = Inception-V3(O);
(H, H̃) = Bi-LSTM(X);
for j = 1 to N do                                  // BICP
  if j == 1 or j == N then
    O'[j] = O[j];
  else
    X'[j] = recommend(H[j-1], H̃[j+1]);
    O'[j] = D'(X'[j]);
  end if
end for
for t = 1 to N do                                  // I2I-cos
  CS = cos (X'[t] , X[t]);
  I2I-cos-CS = I2I-cos-CS + CS;
end for
I2I-cos-CS = Avg(I2I-cos-CS);
output(O', I2I-cos-CS);
```

Fig. 4. Algorithm of BICP with I2I-coss.

Therefore, we propose the method to directly calculate the cosine similarity between two instances that the pre-trained Inception-V3 transforms the feature representations. At each position in the original input outfit, the compatibility score is computed for the generated new outfit. The cosine similarity serves as the compatibility score for each instance. Since an outfit consists of multiple instances, we sum the compatibility scores calculated for each position and take the average to obtain the overall compatibility score for this input entire outfit:

$$E(\Theta_f, \Theta_b) = \frac{1}{N}\sum_{t=1}^{N} cos(X'_t, X_t). \quad (2)$$

where X'_t represents the newly generated instance at the t-th position, transforming into a feature vector using the pre-trained Inception-V3 model on ImageNet, and *cos* represents the cosine similarity. The above method is the compatibility metric approach. It is called I2I-cos, which is in the range of [-1, 1]. In an ideal scenario, the calculated compatibility score is 1 which also shows whether the iteration will converge. To better illustrate the iterative process, we use the following Algorithm 1 (Figure 4) to show the complete pseudocode for an understandable representation. Besides, Figure 5 is an illustrative example referring to the proposed algorithm.

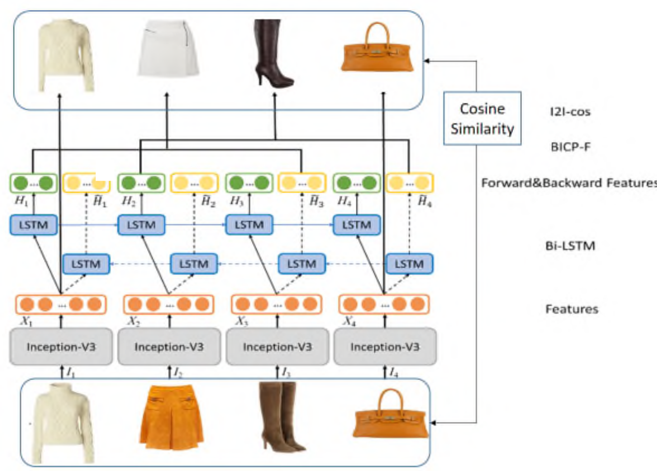

Fig. 5. Architecture of BICP with I2I-coss.

IV. EXPERIMENTS

After describing the proposed method, this section will show the evaluation for the proposed method.

A. *Experimental Data*

The outfit dataset collected by Han et al. from the Polyvore website contains 21,889 outfits and 164,379 items. It is split into 17,316 outfits for training, 1,497 outfits for validation and 3,076 outfits for testing. Each outfit length is between 4 to 8. Each item has the corresponding image, text description, and category (such as jeans and skirts, with 380 kinds of categories in total). To evaluate the performance of our proposed methods, we utilized the fashion compatibility prediction data created by [8], which contains 7076 outfits, of which 3076 are compatible and 4000 are incompatible. Compatible outfits are those that have already been well-matched in the testing set, and their compatibility scores are labeled as ones under the Han et al.'s standard. Incompatible

 ISBN: 978-1-68558-182-4

outfits are created by randomly selecting fashion items from the testing set, and their compatibility scores are labeled as zero.

B. Experimental Settings

To test our proposed method, we directly used the trained model from Han et al.'s approach. We likewise initialized the Inception-V3 parameters to those pre-trained on ImageNet. We extracted a 2048-dimensional (2048D) feature vector for each image using a pre-trained Inception-V3 model and transformed it into a 512D vector as the input to Bi-LSTM using a fully connected layer. The number of hidden layer units in the forward and backward LSTMs was set to 512. We set the number of iterations to 4. The fashion recommendation programs in this paper were implemented in Python, running on a server with NVIDIA Tesla V100 16GB, Intel(R) Xeon(R) Gold 6140 CPU 2.30GHz and 128GB RAM.

C. Experimental Results

Figure 6 shows the distribution diagrams of I2I-cos scores for the original compatible input outfits and the recommended outfits derived by the proposed approach. Comparing the two diagrams, we may find the recommended outfits has higher scores than their input even the original input has been with a high compatibility. This means that the proposed approach can improve the outfit quality in average by the proposed recommendation method BICF.

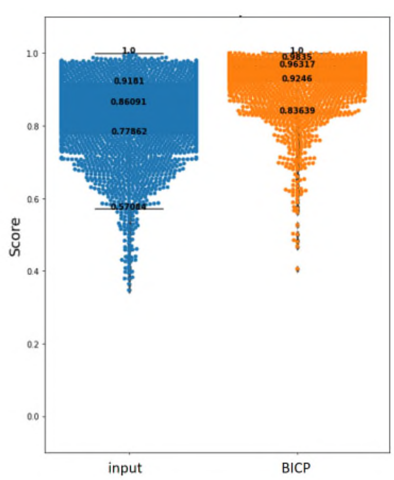

Fig. 6. BICP on compatible outfits evaluated using I2I-cos.

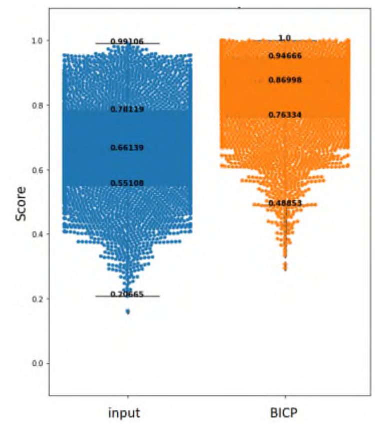

Fig. 7. BICP on incompatible outfits evaluated using I2I-cos.

Figure 7 shows the distribution diagrams of I2I-cos scores for the original incompatible input outfits and the recommended outfits derived by the proposed approach. Since the original input outfits are incompatible, their scores spread wider and are lower in average. In this case, the proposed approach can recommend good outfits and has a significant improvement of the scores. Therefore, the overall score distribution tends to move upwards and become more concentrated.

V. Conclusion

In this paper, we have proposed a novel outfit recommendation mechanism to suggest suitable revisions corresponding to the given category based on outfit inputs of users. The mechanism allows users to input an outfit consisting of a set of their preferred images of items, given which the system will suggest a more suitable one with higher compatibility than the original. We extend Han et al.'s approach by predicting one item, using faster similarity evaluation, and specifying that they are of the same category as the items in the input outfit to form the whole outfit for suggested clothing items. We also employ Han et al.'s compatibility prediction concept to propose a novel evaluation method to evaluate outfit compatibility for the proposed mechanism. Finally, the outfits with higher compatibility scores are recommended. Through the distribution diagrams, it is evident that the proposed outfit recommendation method indeed recommends highly compatible outfits to users. In the future, we will extend this work without fixing the head and tail to reveal the flexibility.

References

[1] W. Chen, et al., “POG: personalized outfit generation for fashion recommendation at Alibaba iFashion,” *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 2662-2670, 2019.

[2] G. Cucurull, P. Taslakian, and D. Vazquez, “Context-aware visual compatibility prediction,” *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12617-12626, 2019.

[3] Z. Cui, Z. Li, S. Wu, X. Y. Zhang, and L. Wang, “Dressing as a whole: outfit compatibility learning based on node-wise graph neural networks,” *Proceedings of the World Wide Web*

Conference, pp. 307-317, 2019.

[4] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional LSTM networks," *Proceedings of 2005 IEEE International Joint Conference on Neural Networks*, Montreal, QC, Canada, pp. 2047-2052, 2005.

[5] X. Han, Z. Wu, Y. G. Jiang, and L. S. Davis, “Learning fashion compatibility with bidirectional LSTMs,” *Proceedings of the 25th ACM International Conference on Multimedia*, pp. 1078-1086, 2017.

[6] Z. Huang, X. Xu, H. Zhu, and M. Zhou, “An efficient group recommendation model with multiattention-based neural networks,” *Journal of IEEE Transactions on Neural Networks and Learning Systems*, Vol. 31, No. 11, pp. 4461-4474, 2020.

[7] R. Iyer, W. Wang and Y. Sun, "Bi-Level Attention Graph Neural Networks," *Proceedings of IEEE International Conference on Data Mining*, pp. 1126-1131, 2021.

[8] P. Kaicheng, Z. Xingxing, and W. K. Wong, “Modeling fashion compatibility with explanation by using bidirectional LSTM,” *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3894-3898, 2021.

[9] W. C. Kang, E. Kim, J. Leskovec, C. Rosenberg, and J. Mcauley, “Complete the look: scene-based complementary product recommendation,” *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10532-10541, 2019.

[10] S. Kumar and M. D. Gupta, “c^+GAN: complementary fashion item recommendation,” *arXiv preprint arXiv:1906.05596*, 2019.

[11] R. Sarkar, et al., “Outfittransformer: outfit representations for fashion recommendation,” *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2263-2267, 2022.

[12] X. Song, et al., “Neurostylist: neural compatibility modeling for clothing matching,” *Proceedings of the 25th ACM International Conference on Multimedia*, pp. 753-761, 2017.

[13] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, “Rethinking the inception architecture for computer vision,” *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2818-2826, 2016.

[14] X. Wang, et al., “Heterogeneous Graph Attention Network,” *Proceedings of the World Wide Web Conference*, Association for Computing Machinery, pp. 2022-2032, 2019.

 ISBN: 978-1-68558-182-4