**POG: Personalized Outfit Generation for Fashion Recommendation at Alibaba iFashion**

Wen Chen, Pipei Huang, Jiaming Xu, Xin Guo, Cheng Guo, Fei Sun, Chao Li, Andreas Pfadler, Huan Zhao, Binqiang Zhao

Alibaba Group, Beijing, China. (Applied Data Science Track, Poster No.5, August 6)

**Introduction**

There exist two requirements for fashion outfit recommendation: the *Compatibility* of the generated fashion outfits, and the *Personalization* in the recommendation process. We attempt to build the bridge between fashion outfit generation and recommendation. For the *Compatibility* requirement, we propose a Fashion Outfit Model (FOM) by setting up a masked item prediction task based on the self-attention mechanism. For the *Personalization* requirement, we propose a Personalized Outfit Generation (POG) model, which uses a Transformer architecture to model both signals from user preference and outfit compatibility. This is the first study to generate personalized outfits based on users’ historical behaviors with an encoder-decoder framework in an industrial-scale application at Alibaba iFashion (Figure 1).

![Figure 1: Alibaba iFashion.](image)

**Dataset**

We provide three datasets describing the outfits, the items, and the user behaviors separately. To the best of our knowledge, our dataset is the largest publicly available dataset of fashion items with rich information compared to existing datasets.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>#Outfits</th>
<th>#Users</th>
<th>#Items</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outfit data</td>
<td>1,013,136</td>
<td>-</td>
<td>583,464</td>
</tr>
<tr>
<td>Item data</td>
<td>-</td>
<td>-</td>
<td>4,747,039</td>
</tr>
<tr>
<td>User data</td>
<td>127,169 3,569,112</td>
<td>4,463,302</td>
<td></td>
</tr>
</tbody>
</table>

Table 1: Statistics of the datasets.

**FOM**

Masking items one at a time in the outfits, we require the model to fill in the blank with the correct item according to the context. Since every item in the outfit is masked to fuse its left and right context, the compatibility between each item and all the other items within the outfit can be learned from the self-attention mechanism. Given an outfit \( F = \{f_1, \ldots, f_t, \ldots, f_n\} \), where \( f_t \) is the \( t \)-th item, we use a particular embedding [MASK] for the masked item. Non-masked items are represented by their multi-modal embeddings. We then represent the set of input embeddings as \( F_{\text{mask}} \).

**POG**

Take advantage of encoder-decoder structure, we aim to translate an user’s historical behaviors to a personalized outfit. Let \( U \) denote the set of all users and \( F \) be the set of all outfits. We use a sequence of user behaviors \( U = \{u_1, \ldots, u_t, \ldots, u_m\} \) to characterize an user, where \( u_t \) are the clicked items by the user. \( F = \{f_1, \ldots, f_t, \ldots, f_n\} \) is the clicked outfit from the same user, where \( f_t \) are the items in the outfit. At each time step, we predict the next outfit item given previous outfit items and user’s click sequence on items \( U \).

We minimize the following loss function:

\[
\mathcal{L}_F = \frac{1}{n_{\text{mask}} - 1} \sum_{i=1}^{n_{\text{mask}}} \log Pr(f_{\text{mask}}|f_{\text{mask}}; \Theta_F)
\]

where \( \Theta_F \) denotes the model parameters, and \( Pr(\cdot) \) is the probability of choosing the correct item conditioned on the non-masked items. The model architecture is shown in Figure 2.

**Didas Platform**

We develop a platform named Dida which is able to generate personalized outfits automatically. Dida is widely used by more than one million operators at Alibaba. About 6 million personalized outfits are generated everyday with high quality. So far, the outfits have been recommended to more than 5.4 million users.

![Figure 4: Dida platform.](image)

**Experiment**

Our model significantly outperforms other alternative methods through outfit compatibility experiments, including pushing the FITB (Fill In The Blanks) benchmark to 68.79% (5.98% relative improvement) and CP (Compatibility Prediction) benchmark to 86.32% (25.81% relative improvement). Through extensive online experiments, we show that POG clearly outperforms the CF method by 70% increase in CTR (Click-Through-Rate) metric. The online cases can be found in Figure 5.

![Figure 5: The online cases of POG.](image)

**Contact Information**

*Email: chenyu.cw@alibaba-inc.com*