Fine-Grained Visual Attribute Extraction from Fashion Wear

Viral Parekh[1]*, Karimulla Shaik[1]*, Soma Biswas[2], Muthusamy Chelliah[1]
[1]Flipkart Internet Private Limited, [2]Indian Institue of Science

Abstract

Automatically extracting visual attributes for e-commerce data has widespread applications in cataloging, catalogue qualification and enrichment, visual search, etc. Here, we address the task of visual attribute extraction for a highly challenging real-world fashion data from Flipkart catalogue (an Indian e-commerce platform), which is collected from seller uploaded product images. This data not only contains widely varying categories (e.g., shirt, sari, shoes), but also has both coarse-grained (e.g., occasion, top type, sari type) and fine-grained (e.g., neck type, print type) attributes. Training examples available for different attributes are highly imbalanced, making this task even more challenging. To this end, we propose an end-to-end framework which integrates multi-task learning with transformer as an attention module, in addition to handling the data imbalance. The proposed architecture supports multiple attributes across various product categories in a scalable manner. Extensive experiments on the in-house dataset shows effectiveness of the proposed framework in improving performance of the fine-grained attributes by 13% on the baseline across the attributes.

1. Introduction

Cataloging is a process of listing products in catalogue which is specific to e-commerce companies. In general, sellers submit product requests with product information such as images, attributes, free text and e-commerce companies do a quality check before listing. There is a trade off between the amount of information to be filled by seller and his/her experience and hence by design some attributes are mandatory and some are optional. This results in majority of the products having partial information in the catalogue. Automatic extraction of attributes from product images not only enriches the catalogue, but also reduces human subjectivity and hence improves consistency among the attributes. This also keeps seller experience intact. In addition, visual attribute extraction enables other applications such as visual search to retrieve matching products, given a visual query. Here, we work with a part of Flipkart catalogue data. The dataset consists 25 product types from fashion category and 48 attributes (having binary or multiple labels) and each attribute has variable number of values which results in a total of approximately 1900 labels. We provide details of the dataset, its challenges and how it is different from existing ones in subsequent sections.

We propose a deep framework which leverages the power of multi-task learning with attention mechanism, along with data-imbalance handling techniques to address this challenging task. Finally, extensive experiments validate effectiveness of the proposed framework.

*indicates equal contribution

2. Related Work

Previous work in attribute extraction falls under the following three broad categories:

Multi-Class Prediction: It has a class label space which is a combination of values of attributes. This formulation is preferred for small label space; however, it does not allow prediction of value combinations not seen during training.

Multi-Label Prediction: Here the value of each attribute is considered to be a different label and thus each input instance is assigned to multiple class labels.

Multi-Task Prediction This is the multi-label model that adopts a multi-class classifier for predicting one or more values for each attribute [2, 8].

Deep CNN features are adapted as representations to learn semantic attributes. Multi-task learning (MTL) in addition is a paradigm that uses other related tasks to improve generalization performance of learning tasks. MTL allows CNN models to simultaneously share visual knowledge among different attribute categories; each CNN will generate attribute-specific representations, learning on which features helps predict attributes [1]. MTL prompts sharing of CNN features between object classification and location tasks [14]. Prior knowledge about attribute relationships is utilized to minimize intra-class (e.g., gender) and maximize inter-class distance. Attention mechanism helps incorporate impact of positions for clothing attribute prediction with only image-level annotations. A novel task-aware attention mechanism [15] estimates importance of each position across different tasks. SAC [4] combines CNNs and self-attention mechanism to represent fine-grained clothing attributes. Based on Grad-CAM [9], this approach helps to visualize which image parts contribute to the prediction result.

Attribute response is location-sensitive, i.e., different spatial locations have various contributions. In [11],[6], the

landmark information is used to generate better feature representation for attribute prediction. Apart from product attributes and landmarks, there is focus on other tasks like localization and segmentation [16]. Along with images, other modalities like text in form of search query and product description are used to improve attribute extraction performance as well [7, 5, 3].

3. Dataset Details

The data set used in this work is obtained from Flipkart catalogue, and consists of 25 products categories uploaded by the sellers in last 2 years. After manual inspection of all the possible attributes of these categories, we have identified 48 visual attributes which are applicable for one or more products. The visual attributes are a mix of fine-grained (e.g., print type, print coverage, neck type) and coarse-grained attributes (e.g., pattern, top wear length, product). Overall, our train, validation and test split consist of 783871, 156770 and 177050 images respectively. This

Figure 1. Few examples of two fine-grained attributes - *neck type* (top 2 rows) and *print type* (bottom 2 rows).

dataset is very challenging in terms of the following: **1. Distribution:** The dataset has heavy imbalance at every level - category, attribute and attribute values. At category level, the smallest one *cargo* has around 2700 data-points while the largest one *t-shirt* has around 2.3 lakhs data-points. At attribute level the *ideal for* is present for almost 100% of the data-points, while many attributes are present for only $1-2\%$ of the data-points. Similar imbalance is observed at attribute value level for all the attributes. **2. Missing Attributes:** For a given product, few attributes are mandatory for the sellers to list, while many others are optional, e.g., color, pattern, product category are mandatory attributes, while pattern coverage, closure type are *good to have* attributes. Thus, the dataset has several missing values, especially the *good to have* attributes. **3. Label distribution:** One attribute might be applicable for multiple products, but the allowed attribute values might have different distribution for different products, e.g., *closure type* attribute has values like *button* and *zip* for jeans, while it has values like *drawstring* and *elastic* for pyjamas, while trousers have all the above values. **4. Fine-grained attributes:** Few of the attributes are extremely fine-grained with large number

Figure 2. Different ways to wear dupatta.

of attribute values. For example pattern, print type, neck type, top type have 89, 91, 50 and 49 attribute values respectively. **5. Cross-product interference:** In most of the images, the main clothing item is worn by human model. So along with the primary product, other clothing items are also visible, e.g., in a trouser image some portion of the t-shirt or shirt is also visible. In some cases, accessories (e.g., dupatta, bags) occlude the main product. These cases are challenging when we want to make prediction for generic attributes like pattern, print type, occasion, etc. **6. Uniqueness:** The dataset used in our work has some unique features and challenges compared to the existing data sets. *Indian fashion categories*: As the data set is collated from an Indian e-commerce database, it contains several categories which are specific to India, e.g., we have categories like sari, kurta, kurti, etc. *Style of wearing*: Most of the western wears have standard way of wearing them. However there are many ways to drape a sari. In Figure 2, we observe different ways in which a dupatta (with ethnic-set) can be worn.

4. Proposed Framework

The proposed end-to-end framework to address the attribute extraction task is discussed next.

4.1. Backbone Network

The overall framework is shown in Figure 3. The input image is passed through the backbone network, which serves as the feature extraction unit. The output of backbone network is the base feature vector, which is the common input for all the branch networks, each dedicated to one attribute. In this work, we have used EfficientNet [12] as the base feature extractor. We have experimented with several other architectures as well, such as VGG16, VGG19, ResNet and Inception. EfficientNet gave $5-6\%$ improvement compared to the other backbones and it is also efficient in terms of parameters, which motivated us to use it as the backbone architecture. In our experiments, we have chosen EfficientNet-B0, which gives $7 \times 7 \times 1280$-d feature vector as output. This is passed through an attention module followed by global average pooling to get the final 1280×1 feature vector, which is passed to the branch networks.

We use a transformer block [13] as attention module with 8 heads; feature maps corresponding to all spatial positions of output of the backbone network (which is 49×1280) is fed to this module.

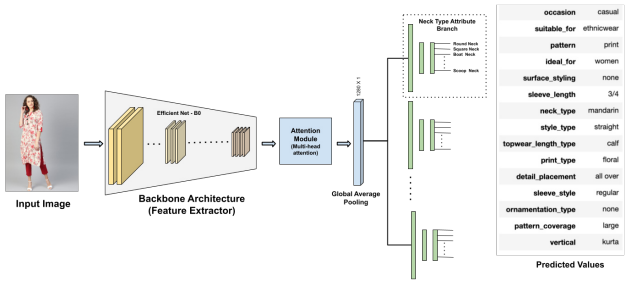

Figure 3. Architecture of the proposed end-to-end framework.

4.2. Branch Network

The base feature vector obtained from the backbone network and attention module is passed through several branch networks. Each branch network caters to each attribute of interest and contains individual trainable classification network. In the data set, there are a total of 49 attributes, which includes the category label (after attribute standardisation and merging), so we have 49 branch networks. Each branch consists of one fully connected layer followed by an output layer of size equal to the number of attribute values for that attribute.

In our final setup, to handle the imbalance at category level, we have done majority down-sampling of the images to reduce its effect. We have considered all the attributes as multi-label attributes and handled mutually exclusive attributes during post processing. We use sigmoid activation at leaf level and only consider labels with score greater than 0.5 as predictions. The loss function for this multi-label classification setup with support to handle the missing label is as follows:

$$loss(Y, \hat{Y}) = \sum_{t=1}^{T} w_t \times BCE^t_{mask}(Y, \hat{Y}) \qquad (1)$$

$$BCE^t_{mask}(Y, \hat{Y}) = \frac{1}{N_t} \sum_{i=1}^{N_t} mask^t_i \sum_{c=1}^{C_t} BCE(y^{tc}_i, \hat{y_i}^{tc})$$

where T is the number of tasks, w_t, C_t, N_t are weight, number of classes and number of samples for task t respectively. n_t is the number of samples that are applicable for a given task t out of N_t. The mask is given by

$$mask^t_i = \begin{cases} 0 & \text{if } y_i < 0 \\ 1 & \text{else} \end{cases}$$

$$BCE = -y^{tc}_i \times log(\hat{y_i}^{tc}) - (1 - y^{tc}_i) \times log(1 - \hat{y_i}^{tc})$$

The argument for BCE ($BCE(y^{tc}_i, \hat{y_i}^{tc})$) is omitted for brivity. Here $\hat{y}^{tc}_i$ is predicted label and y^{tc}_i is the ground truth for task t, class c and sample i. For missing label cases, ground truth values are set as -1.

5. Experiments

In this section, we report the experimental results of the proposed framework on the dataset, starting with the evaluation metrics.

Evaluation Metric: Image attribute extraction is fundamentally an image classification problem. However, depending on nature of the attribute, this can be formulated as either multi-class or multi-label classification problem. Here, a given attribute may not be applicable for all the classes, and for this attribute, these should be considered as negative classes, e.g., *neck type* is not applicable for bottomwear verticals like jeans. In this work, for evaluation, we use multi-label accuracy at sample level and then average over all examples in the validation set [10] as follows:

$$AverageAccuracy = \frac{1}{N_t} \sum_{j=1}^{N_t} \frac{\left|Y_j \cap \hat{Y}_j\right|}{\left|Y_j \cup \hat{Y}_j\right|} \qquad (2)$$

$\hat{Y}$ and Y are the predicted and ground truth labels and N_t is the number of samples. Accuracy for each instance is the proportion of predicted correct labels to total number (predicted and actual) of labels for that instance. In case of mutually exclusive attribute, this metric behaves as simple accuracy.

Handling Missing Labels and Multi-labels: As explained earlier, the data set contains many examples, for which the values are missing for few attributes (mainly good-to-have attributes). This is different from the scenario where the attribute is not applicable for a given category, and thus those attribute values are unavailable. We handle such missing labels so that they do not effect the final loss. Some of the attributes have mutually exclusive attribute values, e.g., *neck type* is a mutually exclusive attribute, where a given product can have only one value, while the same product can have multiple values for the attribute *occasion*, like casual, party wear, etc. One way to handle this is to treat the classification branches for mutually exclusive attributes as multi-class classification problem, while treat the others as multi-class, multi-label classification problem. For a mutually exclusive attribute, one value is predicted regardless of whether that attribute is applicable for the given category or not. e.g. neck type is not applicable for jeans. This leads to unstable model training. In this work, we treat all the classification branches as multi-label, and for mutually exclusive attributes we select the label with highest score (if any) after applying threshold of 0.5. In a multi-label setting, the model can give low score for all the classes, so the issue mentioned in previous approach is not observed.

Implementation Details: Here, we have used EfficientNet-B0 pre-trained with Imagenet dataset as the backbone network. We trained the model end-to-end using the training split of the dataset. In the branch network, there are total 49 classification branches, out of which 48 are for the attributes, and one for category classification. Each branch contains two hidden layers followed by the output layer. The number of nodes in the output layer is the same as the

number of allowed attribute values for that attribute, e.g., *ideal for* attribute has 7 attributes values ('baby boys', 'baby girls', 'boys', 'couple', 'girls', 'men', 'women'), thus there will be 7 nodes in its output layer. The input images are resized to 224×224. We use image augmentations like zoom in and out within range of +/- 20%, translation within +/- 20% both in horizontal and vertical directions, rotation within +/- 20 degrees, shear in range +/-16 and flipping of images from left to right with 50% probability. The model is trained with Adam optimizer, learning rate of $1e-5$ with decay rate of 0.75 for 15 epochs, and 32 batch size.

Experimental Results: We perform both objective and subjective analysis of the proposed framework. For objective measure, we use the accuracy metrics in (2). Figure 4 shows the attribute-wise average accuracy scores. We observe that attributes which are applicable across multiple products (e.g., ideal for, occasion, category type, pattern, pattern coverage) have relatively good scores. Also the attributes which are mandatory for a given category (sleeve details, neck type, sleeve length etc.) and thus have less missing values perform well. We perform ablation study for the attention module and observed that attention is indeed improving performance of several attributes like belt included, closure type, dupatta included, pattern coverage, etc. For the other attributes, we have not seen any significant improvement or degradation.

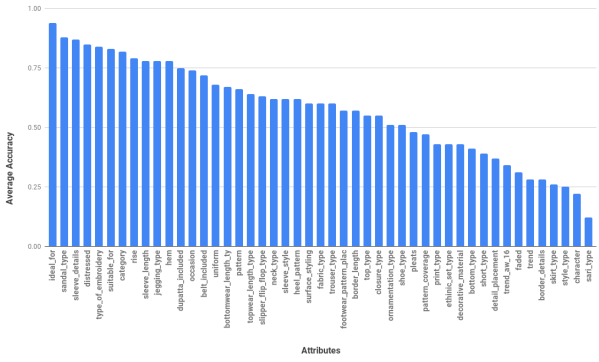

Figure 4. Attribute wise average accuracy scores.

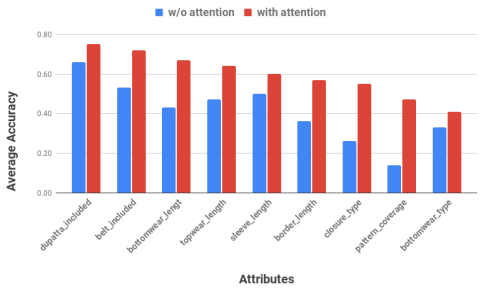

Figure 5. Average accuracy with and without attention module.

Visual Analysis: We analysed the trained model using Grad-Cam [9], which uses the gradients of any target concept flowing into the final convolutional layer to produce a coarse localization map highlighting the important image regions used for predicting the concept. We use the last convolutional layer before global average pooling to generate these visualizations. Figure 6 indicates high activation regions which are suitable for the given attribute, where red denotes region of high importance. In the top left image, predicted value for *suitable for* attribute is "western" and the regions used are neck, shoulder and feet. For the second image (top row), *sleeve style* is predicted as "sleeve-less" for which the model focuses on both hands. Figure 7 shows few images along with the predicted attributes.

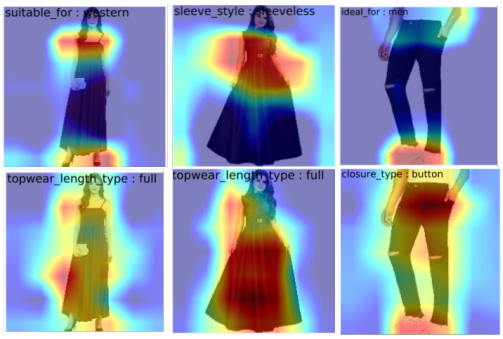

Figure 6. Grad-cam visualisation to illustrate how the model looks at different part of the images to predict different attributes.

Figure 7. Few images along with attributes predicted by our model. All the predictions are selected with confidence higher than 0.5.

6. Conclusion and Future work

Here, we have proposed an end-to-end framework for attribute prediction on a very challenging Indian fashion product data set. We highlight the challenges and uniqueness of the data set compared to the existing ones. We observe that the framework performs satisfactorily for multiple attributes across various product categories in a scalable manner.

Possible future directions of the current work are as follows: **Feature Fusion**: The size of important regions for predicting different attributes may be quite different, thus the receptive field required to make predictions for different attributes can be different. So, we can use different feature layers or fusion of multiple layers before the final classification network specific to each attribute. **Knowledge base**: Many of the attributes and their values are related to each other and have strong positive or negative correlation, e.g., formal shirts usually have full sleeves, and embroidered saris are usually suitable for wedding and festivals. These information can be effectively exploited to make better predictions.

References

[1] Abrar H Abdulnabi, Gang Wang, Jiwen Lu, and Kui Jia. Multi-task cnn model for attribute prediction. *IEEE Transactions on Multimedia*, 17(11):1949–1959, 2015.

[2] Sandeep Singh Adhikari, Sukhneer Singh, Anoop Rajagopal, and Aruna Rajan. Progressive fashion attribute extraction. *arXiv preprint arXiv:1907.00157*, 2019.

[3] Hasan Sait Arslan, Kairit Sirts, Mark Fishel, and Gholamreza Anbarjafari. Multimodal sequential fashion attribute prediction. *Information*, 10, 2019.

[4] Yutong Chun, Chuansheng Wang, and Mingke He. A novel clothing attribute representation network-based self-attention mechanism. *IEEEAccess*, 8:201762–201769, 2020.

[5] Boeun Kim, Young Han Lee, Hyedong Jung, and Choongsang Cho. Distinctive-attribute extraction for image captioning. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pages 0–0, 2018.

[6] 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.

[7] Robert L Logan IV, Samuel Humeau, and Sameer Singh. Multimodal attribute extraction. *arXiv preprint arXiv:1711.11118*, 2017.

[8] Beatriz Quintino Ferreira, Luís Baía, João Faria, and Ricardo Gamelas Sousa. A unified model with structured output for fashion images classification. *arXiv e-prints*, pages arXiv–1806, 2018.

[9] 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.

[10] Mohammad S Sorower. A literature survey on algorithms for multi-label learning. Technical report, 2010.

[11] Haibo Su, Peng Wang, Lingqiao Liu, Hui Li, Zhen Li, and Yanning Zhang. Where to look and how to describe: Fashion image retrieval with an attentional heterogeneous bilinear network. *IEEE Transactions on Circuits and Systems for Video Technology*, 2020.

[12] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International Conference on Machine Learning*, pages 6105–6114. PMLR, 2019.

[13] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *arXiv preprint arXiv:1706.03762*, 2017.

[14] Yizhang Xia, Baitong Chen, Wenjin Lu, Frans Coenen, and Bailing Zhang. Attributes-oriented clothing description and retrieval with multi-task convolutional neural network. In *2017 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*, pages 804–808. IEEE, 2017.

[15] Sanyi Zhang, Zhanjie Song, Xiaochun Cao, Hua Zhang, and Jie Zhou. Task-aware attention model for clothing attribute prediction. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(4):1051–1064, 2019.

[16] Yuwei Zhang, Peng Zhang, Chun Yuan, and Zhi Wang. Texture and shape biased two-stream networks for clothing classification and attribute recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13538–13547, 2020.