Label2Label: A Language Modeling Framework for Multi-Attribute Learning

Wanhua Li, Zhexuan Cao, Jianjiang Feng, Jie Zhou, and Jiwen Lu*

[1] Department of Automation, Tsinghua University, China
[2] Beijing National Research Center for Information Science and Technology, China
{wanhua016,caozx00}@gmail.com; {jfeng,jzhou,lujiwen}@tsinghua.edu.cn

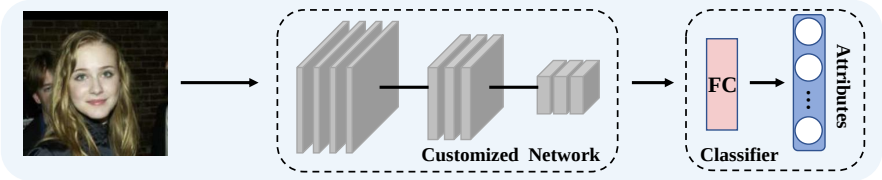
Abstract. Objects are usually associated with multiple attributes, and these attributes often exhibit high correlations. Modeling complex relationships between attributes poses a great challenge for multi-attribute learning. This paper proposes a simple yet generic framework named Label2Label to exploit the complex attribute correlations. Label2Label is the first attempt for multi-attribute prediction from the perspective of language modeling. Specifically, it treats each attribute label as a "word" describing the sample. As each sample is annotated with multiple attribute labels, these "words" will naturally form an unordered but meaningful "sentence", which depicts the semantic information of the corresponding sample. Inspired by the remarkable success of pre-training language models in NLP, Label2Label introduces an image-conditioned masked language model, which randomly masks some of the "word" tokens from the label "sentence" and aims to recover them based on the masked "sentence" and the context conveyed by image features. Our intuition is that the instance-wise attribute relations are well grasped if the neural net can infer the missing attributes based on the context and the remaining attribute hints. Label2Label is conceptually simple and empirically powerful. Without incorporating task-specific prior knowledge and highly specialized network designs, our approach achieves state-of-the-art results on three different multi-attribute learning tasks, compared to highly customized domain-specific methods. Code is available at https://github.com/Li-Wanhua/Label2Label.

Keywords: multi-attribute, language modeling, attribute relations

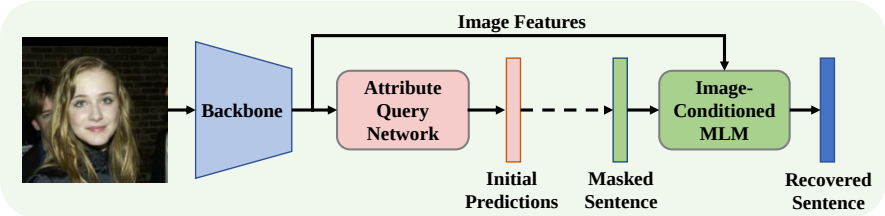
1 Introduction

Attributes are mid-level semantic properties for objects which are shared across categories [14–16, 31]. We can describe objects with a wide variety of attributes. For example, human beings easily perceive gender, hairstyle, expression, and so on from a facial image [30, 32]. Multi-attribute learning, which aims to predict the attributes of an object accurately, is essentially a multi-label classification task [49]. As multi-attribute learning involves many important tasks, including

* Corresponding author



(a) Existing Multi-task Learning Framework



(b) Our Language Modeling Framework

Fig. 1. Comparisons of the existing multi-task learning framework and our proposed language modeling framework.

facial attribute recognition [5,24,38], pedestrian attribute recognition [17,23,51], and cloth attribute prediction [37,59], it plays a central role in a wide range of applications, such as face identification [5], scene understanding [48], person retrieval [28], and fashion search [2].

For a given sample, many of its attributes are correlated. For example, if we observe that a person has blond hair and heavy makeup, the probability of that person being attractive is high. Another example is that the attributes of beard and woman are almost impossible to appear on a person at the same time. Modeling complex inter-attribute associations is an important challenge for multi-attribute learning. To address this challenge, most existing approaches [5,23,45,51] adopt a multi-task learning framework, which formulates multi-attribute recognition as a multi-label classification task and simultaneously learns multiple binary classifiers. To boost the performance, many methods further incorporate domain-specific prior knowledge. For example, PS-MCNN [5] divides all attributes into four groups and presents highly customized network architectures to learn shared and group-specific representations for face attributes. In addition, some methods attempt to introduce additional domain-specific guidance [24] or annotations [37]. However, these methods struggle to model sample-wise attribute relationships with a simple multi-task learning framework.

Recent years have witnessed great progress in the large-scale pre-training language models [4,11,43]. As a representative work, BERT [11] utilizes a masked language model (MLM) [52] to capture the word co-occurrence and language structure. Inspired by these methods, we propose a language modeling framework named Label2Label to model the complex instance-wise attribute relations.

Specifically, we regard an attribute label as a "word", which describes the current state of the sample from a certain point of view. For example, we treat the labels "attractive" and "no eyeglasses" as two "words", which give us a sketch of the sample from different perspectives. As multiple attribute labels of each sample are used to depict the same object, these "words" can be organized as an unordered yet meaningful "sentence". For example, we can describe the human face in Fig. 1 with the sentence "attractive, not bald, brown hair, no eyeglasses, not male, wearing lipstick, ...". Although this "sentence" has no grammatical structure, it can convey some contextual semantic information. By treating multiple attribute labels as a "sentence", we exploit the correlation between attributes with a language modeling framework.

Our proposed Label2Label consists of an attribute query network (AQN) and an image-conditioned masked language model (IC-MLM). The attribute query network first generates the initial attribute predictions. Then these predictions are treated as pseudo label "sentences" and sent to the IC-MLM. Instead of simply adopting the masked language modeling framework, our IC-MLM randomly masks some "word" tokens from the pseudo label "sentence" and predicts the masked "words" conditioned on the masked "sentence" and image features. The proposed image-conditioned masked language model provides partial attribute prompts during the precise mapping from images to attribute categories, thereby facilitating the model to learn complex sample-level attribute correlations. We take facial attribute recognition as an example and show the key differences between our method and existing methods in Fig. 1.

We summarize the contributions of this paper as follows:

- We propose Label2Label to model the complex attribute relations from the perspective of language modeling. As far as we know, Label2Label is the first language modeling framework for multi-attribute learning.
- Our Label2Label proposes an image-conditioned masked language model to learn complex sample-level attribute correlations, which recovers a "sentence" from the masked one conditioned on image features.
- As a simple and generic framework, Label2Label achieves very competitive results across three multi-attribute learning tasks, compared to highly tailored task-specific approaches.

2 Related Work

Multi-Attribute Recognition: Multi-attribute learning has attracted increasing interest due to its broad applications [2, 5, 28]. It involves many different visual tasks [18, 23, 37] according to the object of interest. Many works focus on domain-specific network architectures. Cao *et al.* [5] proposed a partially shared multi-task convolutional neural network (PS-MCNN) for face attribute recognition. The PS-MCNN consists of four task-specific networks and one shared network to learn shared and task-specific representations. Zhang *et al.* [59] proposed Two-Stream Networks for clothing classification and attribute recognition. Since some attributes are located in the local area of the image, many

methods [17,46,51] resort to the attention mechanism. Guo *et al.* [17] presented a two-branch network and constrained the consistency between two attention heatmaps. A multi-scale visual attention and aggregation method was introduced in [46], which extracted visual attention masks with only attribute-level supervision. Tang *et al.* [51] proposed a flexible attribute localization module to learn attribute-specific regional features. Some other methods [24,37] further attempt to use additional domain-specific guidance. Semantic segmentation was employed in [24] to guide the attention of the attribute prediction. Liu *et al.* [37] learned clothing attributes with additional landmark labels. There are also some methods [50,60] to study multi-attribute recognition with insufficient data, but this is beyond the scope of this paper.

Language Modeling: Pre-training language models is a foundational problem for NLP. ELMo [43] was proposed to learn deep contextualized word representations. It was trained with a bidirectional language model objective, which combined both a forward and backward language model. ELMo representations significantly improve the performance across six NLP tasks. GPT [44] employed a standard language model objective to pre-train a language model on large unlabeled text corpora. The Transformer was used as the model architecture. The pre-trained model was fine-tuned on downstream tasks and achieved excellent results in 9 of 12 tasks. BERT [11] used a masked language model pre-training objective, which enabled BERT to learn bidirectional representations conditioned on the left and right context. BERT employed a multi-layer bidirectional Transformer encoder and advanced the state-of-the-art performance. Our work is inspired by the recent success of these methods and is the first attempt to model multi-attribute learning from the perspective of language modeling.

Transformer for Computer Vision: Transformer [53] was first proposed for sequence modeling in NLP. Recently, Transformer-based methods have been deployed in many computer vision tasks [3,19,36,42,54,56,57]. ViT [13] demonstrated that a pure transformer architecture achieved very competitive results on image classification tasks. DETR [6] formulated the object detection as a set prediction problem and employed a transformer encoder-decoder architecture. Pix2Seq [8] regarded object detection as a language modeling task and obtained competitive results. Zheng *et al.* [61] replaced the encoder of FCN with a pure transformer for semantic segmentation. Liu *et al.* [34] utilized the Transformer decoder architecture for multi-label classification. Temporal query networks were introduced in [57] for fine-grained video understanding with a query-response mechanism. There are also some efforts [9,25,41] to apply Transformer to the task of multi-label image classification. Note that the main contribution of this paper is not the use of Transformer, but modeling multi-attribute recognition from the perspective of language modeling.

3 Approach

In this section, we first give an overview of our framework. Then we present the details of the proposed attribute query network and image-conditioned masked

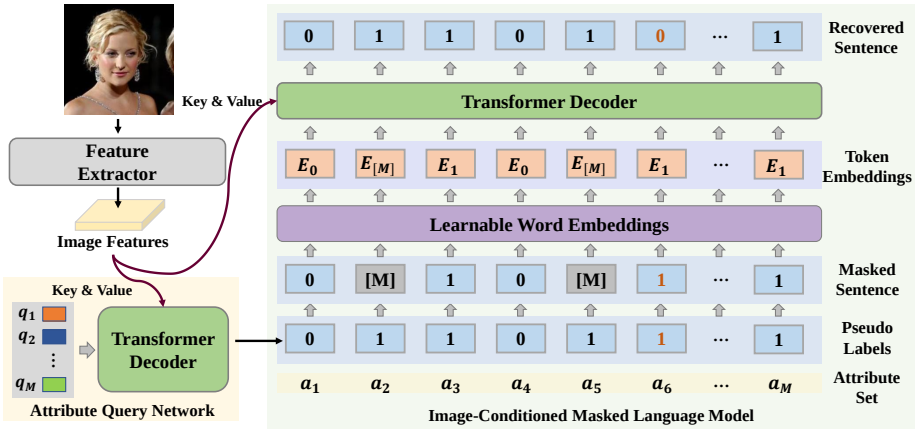


Fig. 2. The pipeline of our framework. We recover the entire label "sentence" with a Transformer decoder module, which is conditioned on the token embeddings and image features. Although there are some wrong "words" in the pseudo labels, which are shown in orange, we can treat them as another form of masks. Here $\boldsymbol{E_1}$ or $\boldsymbol{E_0}$ indicates the presence or absence of an attribute.

language model. Lastly, we introduce the training objective function and inference process of our method.

3.1 Overview

Given a sample $\boldsymbol{x}$ from a dataset $\mathcal{D}$ with M attribute types, we aim to predict the multiple attributes $\boldsymbol{y}$ to the image $\boldsymbol{x}$. We let $\mathcal{A} = \{\boldsymbol{a}_1, \boldsymbol{a}_2, ..., \boldsymbol{a}_M\}$ denote the attribute set, where $\boldsymbol{a}_j (1 \leq j \leq M)$ represents the j-th attribute type. For simplicity, we assume that the values of all attribute types are binary. In other words, the value of $\boldsymbol{a}_j$ is 0 or 1, where 1 means that the sample has this attribute and 0 means not. However, our method can be easily extended to the case where each attribute type is multi-valued. With this assumption, we have $\boldsymbol{y} \in \{0, 1\}^M$. Existing methods [5, 23] usually employ a multi-tasking learning framework, which uses M binary classifiers to predict M attributes respectively. Binary cross-entropy loss is used as the objective.

This paper proposes a language modeling framework. We show the pipeline of our framework in Fig. 2. The key idea of this paper is to treat attribute labels as unordered "sentences" and use an image-conditioned masked language model to exploit the relationships between attributes. Although we can directly use the real attribute labels as the input of the IC-MLM during training, we cannot access these labels for inference. To address this issue, our Label2Label introduces an attribute query network to generate the initial attribute predictions. These predictions are then treated as pseudo-labels and used as input to the IC-MLM in the training and testing phases.

3.2 Attribute Query Network

Given an input image $\boldsymbol{x} \in \mathbb{R}^{H_0 \times W_0 \times 3}$ and its corresponding label $\boldsymbol{y} = \{y_j | 1 \leq j \leq M\}$, we send the image to a feature extractor to obtain the image features, where H_0 and W_0 denote the height and width of the input image respectively, y_j denotes the value of j-th attribute $\boldsymbol{a}_j$ for the sample $\boldsymbol{x}$. As our framework is agnostic to the feature extractor, we can use any popular backbones such as ResNet-50 [20] and ViT [13]. A naive way to generate initial attribute predictions is to directly feed the extracted image features to a linear layer and learn M binary classifiers. As recent progress [12, 25, 34, 57] shows the superiority of Transformer, we consider using the Transformer decoder to implement our attribute query network to generate initial predictions with higher quality.

Our attribute query network learns a set of permutation-invariant query vectors $\boldsymbol{Q} = \{\boldsymbol{q}_1, \boldsymbol{q}_2, ..., \boldsymbol{q}_M\}$, where each query $\boldsymbol{q}_j$ corresponds to an attribute type $\boldsymbol{a}_j$. Then each query vector $\boldsymbol{q}_j$ pools the attribute-related features from the image features with Transformer decoder layers and generates the corresponding response vector $\boldsymbol{r}_j$. Finally, we learn a binary classifier for each response vector to generate the initial attribute predictions.

Since many attributes are only located in some local areas of the image, using global image features is not an excellent choice. Therefore, we preserve the spatial dimensions of image features following [34]. For ResNet-50, we simply abandon the global pooling layer and employ the output of the last convolution block as the extracted features. We denote the extracted features as $\boldsymbol{X} \in \mathbb{R}^{H \times W \times d}$, where H, W, and d represent the height, width, and channel of the image features respectively. To fit with the Transformer decoder, we reshape the feature to be $\boldsymbol{X'} \in \mathbb{R}^{HW \times d}$. Following common practices [6, 13], we add 2D-aware position embeddings $\boldsymbol{X}_{pos} \in \mathbb{R}^{HW \times d}$ to the feature vectors $\boldsymbol{X'}$ to retain positional information. In this way, we obtain the visual feature vectors $\widetilde{\boldsymbol{X}} = \boldsymbol{X'} + \boldsymbol{X}_{pos}$.

With the local visual contexts $\widetilde{\boldsymbol{X}}$, the query features $\boldsymbol{Q} = \{\boldsymbol{q}_j \in \mathbb{R}^d | 1 \leq j \leq M\}$ are updated using multi-layer Transformer decoders. Formally, we update the query features $\boldsymbol{Q}_{i-1}$ in the i-th Transformer decoder layer as follows:

$$\begin{aligned} \boldsymbol{Q}_{i-1}^{sa} &= \text{MultiHead}(\boldsymbol{Q}_{i-1}, \boldsymbol{Q}_{i-1}, \boldsymbol{Q}_{i-1}), \\ \boldsymbol{Q}_{i-1}^{ca} &= \text{MultiHead}(\boldsymbol{Q}_{i-1}^{sa}, \widetilde{\boldsymbol{X}}, \boldsymbol{X'}), \\ \boldsymbol{Q}_i &= \text{FFN}(\boldsymbol{Q}_{i-1}^{ca}), \end{aligned} \tag{1}$$

where the MultiHead() and FFN() denote the multi-head attention layer and feed-forward layer respectively. Here we set $\boldsymbol{Q}$ as $\boldsymbol{Q}_0$. The design philosophy is that for each attribute query vector, it can give high attention scores to the interested local visual features to produce attribute-related features. This design is compatible with the locality of some attributes. Assuming that the attribute query network consists of L layers of Transformer decoders, then we denote $\boldsymbol{Q}_L$ as $\boldsymbol{R} = \{\boldsymbol{r}_1, \boldsymbol{r}_2, ..., \boldsymbol{r}_M\}$, where each response vector $\boldsymbol{r}_j \in \mathbb{R}^d$ corresponds to a query vector $\boldsymbol{q}_j$. With the response vectors, we use M independent binary classifiers to predict the attribute values $l_j = \sigma(\boldsymbol{W}_j^T \boldsymbol{r}_j + b_j)$, where $\boldsymbol{W}_j \in \mathbb{R}^d$ and $b_j \in \mathbb{R}^1$ are learnable parameters of the j-th attribute classifier, $\sigma(\cdot)$ is the

sigmoid function and l_j is the predicted probability for attribute $\boldsymbol{a}_j$ of image $\boldsymbol{x}$. In the end, we read out the pseudo label "sentence" $\boldsymbol{s} = \{s_1, \boldsymbol{s}_2, ..., \boldsymbol{s}_M\}$ from the predictions $\{l_j\}$ with $\boldsymbol{s}_j = \mathbb{I}(l_j > 0.5)$, where $\mathbb{I}(\cdot)$ is an indicator function.

It is worth noting that the predictions from the attribute query network are not 100% correct, resulting in some wrong "words" in the generated label "sentence". However, we can treat the wrong "words" as another form of masks, because the wrong predictions account for only a small proportion. In fact, the masking strategy of the wrong word is artificially performed in some language models, such as BERT [11].

3.3 Image-Conditioned Masked Language Model

In existing multi-attribute databases, images are annotated with a variety of attribute labels. This paper is dedicated to modeling sample-wise complex attribute correlations. Instead of treating attribute labels as numbers, we regard them as "words". Since different attribute labels describe the object in an image from different perspectives, we can group them as a sequence of "words". Although the sequence is essentially an unordered "sentence" without any grammatical structure, it still conveys meaningful contextual information. In this way, we treat $\boldsymbol{y}$ as an unordered yet meaningful "sentence", where y_j is a "word".

By treating the labels as sentences, we resort to language modeling methods to mine the instance-level attribute relations effectively. In recent years, pre-training large-scale task-agnostic language models have substantially advanced the development of NLP, among which representative works include ELMo [43], GPT-3 [4], BERT [11], and so on. Inspired by the success of these methods, we consider a masked language model to learn the relationship between "words". We mask some percentage of the attribute label "sentence" $\boldsymbol{y}$ at random, and then reconstruct the entire label "sentence". Specifically, for a binary label sequence, we replace those masked "words" with a special work token `[mask]` to obtain the masked sentence. Then we input the masked sentence to a masked language model, which aims to recover the entire label sequence. While the MLM has proven to be an effective tool in NLP, directly using it for multi-attribute learning is not feasible. Therefore, we propose several important improvements.

Instance-wise Attribute Relations: MLM essentially constructs the task $P(y_1, y_2, ..., y_M | \mathcal{M}(y_1), \mathcal{M}(y_2), ..., \mathcal{M}(y_M))$ to capture the "word" co-occurrence and learn the joint probability of "word" sequences $P(y_1, y_2, ..., y_M)$, where $\mathcal{M}()$ denotes the random masking operation. Such a naive approach leads to two problems. The first problem is that MLM only captures statistical attribute correlations. A diverse dataset means that the mapping $\{\mathcal{M}(y_1), \mathcal{M}(y_2), ..., \mathcal{M}(y_M)\} \mapsto \{y_1, y_2, ..., y_M\}$ is a one-to-many mapping. Therefore MLM only learns how different attributes are statistically related to each other. Meanwhile, our experiments find that this prior can be easily modeled by the attribute query network $P(y_1, y_2, ..., y_M | \boldsymbol{x})$. The second problem is that MLM and attribute query network cannot be jointly trained. Since MLM uses only the hard prediction of the attribute query network, the gradient from MLM cannot influence the training

of the attribute query network. In this way, the method becomes a two-stage label refinement process, which significantly reduces the optimization efficiency.

To address these issues, we propose an image-conditioned masked language model to learn instance-wise attribute relations. Our IC-MLM captures the relations by constructing a task $P(y_1, y_2, ..., y_M|\boldsymbol{x}, \mathcal{M}(y_1), \mathcal{M}(y_2), ..., \mathcal{M}(y_M))$. Introducing an extra image condition is not trivial, as this fundamentally changes the behavior of MLM. With the conditions of image $\boldsymbol{x}$, the transformation $\{\boldsymbol{x}, \mathcal{M}(y_1), \mathcal{M}(y_2), ..., \mathcal{M}(y_M)\} \mapsto \{y_1, y_2, ..., y_M\}$ is an accurate one-to-one mapping. Our IC-MLM infers other attribute values by combining some attribute label prompts and image contexts in the precise image-to-label mapping, which facilitates the model to learn sample-level attribute relations. In addition, IC-MLM and the attribute query network can use shared image features, which enables them to be jointly optimized with a one-stage framework.

Word Embeddings: It is known that the word id is not a good word representation in NLP. Therefore, we need to map the word id to a token embedding. Instead of utilizing existing word embeddings with a large token vocabulary like BERT [11], we directly learn attribute-related word embeddings $\boldsymbol{E}$ from scratch. We use the word embedding module to map the "word" in the masked sentence to the corresponding token embedding. Since all attributes are binary, we need to build a token vocabulary with a size of $2M$ to model all possible attribute words. Also, we need to include the token embedding for the special word `[mask]`. This paper considers three different strategies for the `[mask]` token embedding. The first strategy believes the `[mask]` words for different attributes have different meanings, so M attribute-specific learnable token embeddings are learned, where one `[mask]` token embedding corresponds to one attribute. The second strategy treats the `[mask]` words for different attributes as the same word. Only one attribute-agnostic learnable token embedding is learned and shared by all attributes. The third strategy is based on the second strategy, which simply replaces the learnable token embedding with a fixed $\mathbf{0}$ vector. Our experiments find all three strategies work well while the first strategy performs best.

As mentioned earlier, we use pseudo labels $\boldsymbol{s} = \{s_1, \boldsymbol{s}_2, ..., \boldsymbol{s}_M\}$ as input to IC-MLM, so we actually construct $P(y_1, y_2, ..., y_M|\boldsymbol{x}, \mathcal{M}(\boldsymbol{s}_1), \mathcal{M}(\boldsymbol{s}_2), ..., \mathcal{M}(\boldsymbol{s}_M))$ as the task. We randomly mask out some "words" in the pseudo-label sequence with a probability of α to generate masked label "sentences". The "word" $\mathcal{M}(\boldsymbol{s}_j)$ in the masked label "sentences" may have three values: 0, 1, and `[mask]`. We use the word embedding module to map the masked labels "sentences" to a sequence of token embeddings $\boldsymbol{E} = \{\boldsymbol{E}_1, \boldsymbol{E}_2, ..., \boldsymbol{E}_M\}$ according to the word value, where $\boldsymbol{E}_j \in \mathbb{R}^d$ denotes the embedding for "word" $\mathcal{M}(\boldsymbol{s}_j)$.

Positional Embeddings: In BERT, the positional embedding of each word is added to its corresponding token embeddings to obtain the position information. Since our "sentences" are unordered, there is no need to introduce positional embeddings to "word" representations. We conducted experiments with positional embeddings by randomly defining some word order and found no improvement. Therefore we do not use positional embeddings for "word" representations and the learned model is permutation invariant for "words".

Architecture: In NLP, Transformer encoder layers are usually used to implement MLM, while we use multi-layer Transformer decoders to implement IC-MLM due to additional image input conditions. Following the design philosophy similar to the attribute query network, token embeddings $\boldsymbol{E}$ pool features from the local visual features $\boldsymbol{X}'$ with a cross-attention mechanism. We update the token features $\boldsymbol{E}_{i-1}$ in the i-th Transformer decoder layer as follows:

$$\begin{aligned}\boldsymbol{E}_{i-1}^{sa} &= \text{MultiHead}(\boldsymbol{E}_{i-1}, \boldsymbol{E}_{i-1}, \boldsymbol{E}_{i-1}),\\ \boldsymbol{E}_{i-1}^{ca} &= \text{MultiHead}(\boldsymbol{E}_{i-1}^{sa}, \widetilde{\boldsymbol{X}}, \boldsymbol{X}'),\\ \boldsymbol{E}_{i} &= \text{FFN}(\boldsymbol{E}_{i-1}^{ca}).\end{aligned} \tag{2}$$

We set $\boldsymbol{E}$ to $\boldsymbol{E}_0$ and the number of Transformer decoder layers in IC-MLM to D. Then we denote $\boldsymbol{E}_D$ as $\boldsymbol{R}' = \{\boldsymbol{r}'_1, \boldsymbol{r}'_2, ..., \boldsymbol{r}'_M\}$, where $\boldsymbol{r}'_j$ corresponds to the updated feature of token $\boldsymbol{E}_j$. In the end, we perform the final multi-attribute classification with linear projection layers. Formally, we have:

$$p_j = \sigma(\boldsymbol{W}_j'^T \boldsymbol{r}'_j + b'_j), 1 \leq j \leq M, \tag{3}$$

where $\boldsymbol{W}'_j \in \mathbb{R}^d$ and $b'_j \in \mathbb{R}^1$ are the learnable parameters of the j-th attribute classifier, and p_j is the final predicted probability for attribute $\boldsymbol{a}_j$ of image $\boldsymbol{x}$. Note that we are committed to recovering the entire label "sentence" and not just the masked part. In this reconstruction process, we expect our model to grasp the instance-level attribute relations.

3.4 Objective and Inference

As commonly used in most existing methods [23, 38, 46], we adopt the binary cross-entropy loss to train the IC-MLM. On the other hand, since most of the datasets for multi-attribute recognition are highly imbalanced, different tasks usually use different weighting strategies. The loss function for the IC-MLM is formulated as $\mathcal{L}_{mlm}(\boldsymbol{x}) = \sum_{j=1}^{M} w_j(y_j \log(p_j) + (1-y_j)\log(1-p_j))$, where w_j is the weighting coefficient. According to different tasks, we choose different weighting strategies and always follow the most commonly used strategy for a fair comparison. Meanwhile, to ensure the quality of the generated pseudo label sequences, we also supervise the attribute query network with the same loss function $\mathcal{L}_{aqn}(\boldsymbol{x}) = \sum_{j=1}^{M} w_j(y_j \log(l_j) + (1-y_j)\log(1-l_j))$. The final loss function $\mathcal{L}_{total}$ is a combination of the two loss functions above:

$$\mathcal{L}_{total}(\boldsymbol{x}) = \mathcal{L}_{aqn}(\boldsymbol{x}) + \lambda \mathcal{L}_{mlm}(\boldsymbol{x}), \tag{4}$$

where λ is used to balance these two losses. At inference time, we ignore the masking step and directly input the pseudo label "sentence" to the IC-MLM. Then the output of the IC-MLM is used as the final attribute prediction.

4 Experiments

In this section, we conducted extensive experiments on three multi-attribute learning tasks to validate the effectiveness of the proposed framework.

Table 1. Results with different Transformer decoder layers D for IC-MLM. We fix L as 1.

D	1	2	3	4
Error(%)	12.58	**12.49**	12.54	12.52

Table 2. Results with different Transformer decoder layers L for attribute query network. We fix D as 2.

L	1	2	3	4
Error(%)	**12.49**	12.52	12.50	12.58

Table 3. Results on the LFWA dataset with different mask ratios α.

α	0	0.1	0.15	0.2	0.3
Error(%)	12.55	**12.49**	12.55	12.54	12.57

Table 4. Results on the LFWA dataset with different coefficients λ.

λ	0.5	0.8	1	1.2	1.5
Error(%)	12.64	12.56	**12.49**	12.60	12.63

4.1 Facial Attribute Recognition

Dataset: LFWA [38] is a popular unconstrained facial attribute dataset, which consists of 13,143 facial images of 5,749 identities. Each facial image has 40 attribute annotations. Following the same evaluation protocol in [5, 18, 38], we partition the LFWA dataset into two sets, with 6,263 images for training and 6,880 for testing. All images are pre-cropped to a size of 250 × 250. We adopt the classification error for evaluation following [5, 50].

Experimental Settings: We trained our model for 57 epochs with a batch size of 16. For optimization, we used an SGD optimizer with a base learning rate of 0.01 and cosine learning rate decay. The weight decay was set to 0.001. To augment the dataset, Rand-Augment [10] and Random horizontal flipping were performed. We also adopted Mixup [58] for regularization.

Parameters Analysis: We first analyze the influence of the number of Transformer decoder layers in the attribute query network and IC-MLM. The results are shown in Tables 1 and 2. We see that the best performance is achieved when $L = 1$ and $D = 2$. We further conduct experiments with different mask ratios α and list the results in Table 3. As we mentioned above, the wrong "words" in the pseudo label sequences also provide some form of masks. Therefore, our method performs well when $\alpha = 0$. We observe that our method attains the best performance when $\alpha = 0.1$. Table 4 shows the results with different λ, and we see that $\lambda = 1$ gives the best trade-off in (4). We consider three different strategies for `[MASK]` token embedding and list the results in Table 6. We see that the attribute-specific strategy achieves the best performance among them, as it better models the differences between the attributes. Unless explicitly mentioned, we adopt these optimal parameters in all subsequent experiments.

Ablation Study: To validate the effectiveness of our Label2Label, we also conduct experiments on the LFWA dataset with two baseline methods. We first consider the Attribute Query Network (AQN) method, which ignores the IC-MLM and treats the outputs of AQN in Fig. 2 as final predictions. FC Head method further replaces the Transformer decoder layers in AQN with a linear classification layer. To further verify the generalization of our method, we use dif-

Table 5. Ablation experiments with different backbones.

Backbone	ResNet-50		ResNet-101		ViT-B	
Metric	Error(%)	MACs(G)	Error(%)	MACs(G)	Error(%)	MACs(G)
FC Head	13.63±0.02	5.30	13.05±0.03	10.15	13.73± 0.02	16.85
AQN	13.36±0.04	5.63	12.70±0.02	10.48	13.32±0.04	16.97
Label2Label	**12.49**±0.02	6.30	**12.44**±0.04	11.16	**12.79**±0.01	17.23

Table 6. Results of different strategies for `[Mask]` embeddings.

Strategy	Error(%)
0 Vector	12.60
Attribute-Agnostic	12.57
Attribute-Specific	**12.49**

Table 7. Comparisons of MLM and IC-MLM.

Method	Architecture	Co-training with AQN	Error(%)
MLM	MLP	✗	13.34
	TransEncoder	✗	13.32
IC-MLM	TransDecoder	✗	13.01
	TransDecoder	✓	**12.49**

ferent feature extraction backbone networks for ablation experiments. To better demonstrate the significance of the results, we also report the standard deviation. The results are presented in Table 5. In addition, we report the computation cost (MACs) of each method in Table 5. We observe that our method significantly outperforms FC Head and AQN across various backbones with marginal computational overhead, which illustrates the effectiveness of our method.

We then conducted experiments to show how image-conditioned MLM improves performance. The results are listed in Table 7. As we analyzed above, MLM leads to a two-stage label refinement process. We consider two network architectures to implement MLM: Transformer encoder and multilayer perceptron (MLP). The results show that none of them improve the performance of AQN (13.36%). The reason is that MLM only learns statistical attribute relations, and this prior is easily captured by AQN. Meanwhile, our IC-MLM learns instance-wise attribute relations. To see the benefits of the additional image conditions, we still adopt the two-stage label refinement process, and train Transformer decoder layers with fixed image features. We see that performance is boosted to 13.01%, which demonstrates the effectiveness of modeling instance-

Table 8. Performance comparison with state-of-the-art methods on the LFWA dataset. We report the average classification error results. * indicates that additional labels are used for training, such as identity labels or segment annotations.

Method	Error(%)	Year
SSP + SSG [24]*	12.87	2017
He *et al.* [21]	14.72	2018
AFFAIR [29]	13.87	2018
GNAS [22]	13.63	2018
PS-MCNN [5]*	12.64	2018
DMM-CNN [39]	13.44	2020
SSPL [50]	13.47	2021
Label2Label	**12.49**±0.02	-

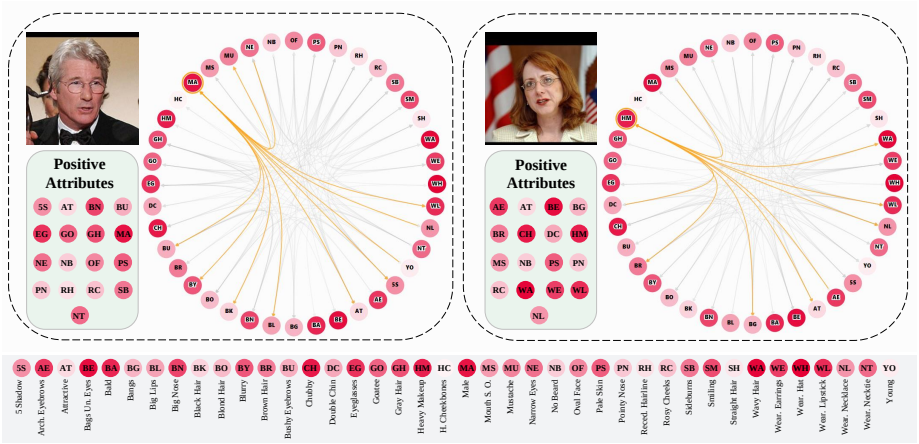


Fig. 3. Visualization of attention scores among attributes in the self-attention layer. We show the attention of the first head at layer 1 with two samples. The positive attributes of each sample are listed in the corresponding bottom-left corner.

wise attribute relations. We further jointly train the IC-MLM and attribute query network, which achieves significant performance improvement. These results illustrate the superiority of the proposed IC-MLM.

Comparison with State-of-the-art Methods: Following [50], we employ ResNet50 as the backbone. We present the performance comparison on the LFWA dataset in Table 8. We observe that our method attains the best performance with a simple framework compared to highly tailored domain-specific methods. Label2Label even exceeds the methods [5, 24] of using additional annotations, which further illustrates the effectiveness of our framework.

Visualization: As the Transformer decoder architecture is used to model the instance-level relations, our method can give better interpretable predictions. We visualize the attention scores in the IC-MLM with DODRIO [55]. As shown in Fig. 3, we see that related attributes tend to have higher attention scores.

4.2 Pedestrian Attribute Prediction

Dataset: The PA-100K [35] dataset is the largest pedestrian attribute dataset so far [51]. It contains 100,000 pedestrian images from 598 scenes, which are collected from real outdoor surveillance videos. All pedestrians in each image are annotated with 26 attributes including gender, handbag, and upper clothing. The dataset is randomly split into three subsets: 80% for training, 10% for validation, and 10% for testing. Following SSC [23], we merge the training set and the validation set for model training. We use five metrics: one label-based and four instance-based. For the label-based metric, we adopt the mean accuracy (mA) metric. For instance-based metrics, we employ accuracy, precision, recall,

Table 9. Comparisons on the PA100K dataset. * represents the reimplementation performance using the same setting. We also report the standard deviations.

Method	mA	Accuracy	Precision	Recall	F1
DeepMAR [26]	72.70	70.39	82.24	80.42	81.32
HPNet [35]	74.21	72.19	82.97	82.09	82.53
VeSPA [47]	76.32	73.00	84.99	81.49	83.20
LGNet [33]	76.96	75.55	86.99	83.17	85.04
PGDM [27]	74.95	73.08	84.36	82.24	83.29
MsVAA [46]*	80.10	76.98	86.26	85.62	85.50
VAC [17]*	79.04	78.95	**88.41**	86.07	86.83
ALM [51]*	79.26	78.64	87.33	86.73	86.64
SSC [23]	81.87	78.89	85.98	**89.10**	86.87
FC Head	77.96±0.06	75.86±0.79	86.27±0.13	84.16±1.02	84.72±0.55
AQN	80.89±0.08	78.51±0.08	86.15±0.40	87.85±0.43	86.58±0.03
Label2Label	**82.24**±0.13	**79.23**±0.13	86.39±0.32	88.57±0.20	**87.08**±0.08

and F1 score. As mentioned in [51], mA and F1 score are more appropriate and convincing criteria for class-imbalanced pedestrian attribute datasets.

Experimental Settings: Following the state-of-the-art methods [17, 23], we adopted ResNet50 as the backbone network to extract image features. We first resize all images into 256×192 pixels. Then random flipping and random cropping were used for data augmentation. SGD optimizer was utilized with the weight decay of 0.0005. We set the initial learning rate of the backbone to 0.01. For fast convergence, we set the initial learning rate of the attribute query network and IC-MLM to 0.1. The batch size was equal to 64. We trained our model for 25 epochs using a plateau learning rate scheduler. We reduced the learning rate by a factor of 10 once learning stagnates and the patience was 4.

Results and Analysis: We report the results in Table 9. We observe that Label2Label achieves the best performance in mA, Accuracy, and F1 score. Compared to the previous state-of-the-art method SSC [23], which designs complex SPAC and SEMC modules to extract discriminative semantic features, our method achieves 0.37% performance improvements in mA. In addition, we report the re-implemented results of the MsVAA, VAC, and ALM methods in the same setting as did in [23]. Our method consistently outperforms these methods. We further show the results of the FC Head and Attribute Query Network. We see that the performance is improved by replacing the FC head with Transformer decoder layers, which shows the superiority of our attribute query network. Our Label2Label outperforms the attribute query network method by 1.35% for mA, which shows the effectiveness of the language modeling framework.

4.3 Clothing Attribute Recognition

Dataset: Clothing Attributes Dataset [7] consists of 1,856 images that contain clothed people. Each image is annotated with 26 clothing attributes, such as

Table 10. The comparisons between our method and other state-of-the-art methods on the Clothing Attributes Dataset. We report accuracy and standard deviation.

Method	Colors	Patterns	Parts	Appearance	Total
S-CNN [1]	90.50	92.90	87.00	89.57	90.43
M-CNN [1]	91.72	94.26	87.96	91.51	91.70
MG-CNN [1]	93.12	95.37	88.65	91.93	92.82
Meng *et al.* [40]	91.64	96.81	89.25	89.53	92.39
FC Head	91.39±0.23	96.07±0.05	87.00±0.27	88.21±0.36	91.57±0.12
AQN	91.98±0.25	96.37±0.23	88.19±0.47	89.89±0.33	92.29±0.05
Label2Label	92.73±0.07	96.82±0.02	88.20±0.09	90.88±0.18	**92.87±0.03**

colors and patterns. We use 1,500 images for training and the rest for testing. For a fair comparison, we only use 23 binary attributes and ignore the remaining three multi-class value attributes as in [1, 40]. We adopt accuracy as the metric and also report the accuracy of four clothing attribute groups following [1, 40].

Experimental Settings: For a fair comparison, we utilized AlexNet to extract image features following [1, 40]. We trained our model for 22 epochs using a cosine decay learning rate scheduler. We utilized an SGD optimizer with an initial learning rate of 0.05. The batch size was set to 32. For the attribute query network, we employed a 2-layer Transformer decoder ($L = 2$).

Results and Analysis: Table 10 shows the results. We observe that our Label2Label attains a total accuracy of 92.87%, which outperforms other methods with a simple framework. MG-CNN learns one CNN for each attribute, resulting in more training parameters and longer training time. Compared with the attribute query network method, our method achieves better performance on all attribute groups, which illustrates the superiority of our framework.

5 Conclusions

In this paper, we have presented Label2Label, which is a simple and generic framework for multi-attribute learning. Different from the existing multi-task learning framework, we proposed a language modeling framework, which regards each attribute label as a "word". Our model learns instance-level attribute relations by the proposed image-conditioned masked language model, which randomly masks some "words" and restores them based on the remaining "sentence" and image context. Compared to well-optimized domain-specific methods, Label2Label attains competitive results on three multi-attribute learning tasks.

Acknowledgments. This work was supported in part by the National Key Research and Development Program of China under Grant 2017YFA0700802, in part by the National Natural Science Foundation of China under Grant 62125603 and Grant U1813218, in part by a grant from the Beijing Academy of Artificial Intelligence (BAAI). The authors would sincerely thank Yongming Rao and Zhiheng Li for their generous helps.

References

1. Abdulnabi, A.H., Wang, G., Lu, J., Jia, K.: Multi-task cnn model for attribute prediction. TMM **17**(11), 1949–1959 (2015)
2. Ak, K.E., Kassim, A.A., Lim, J.H., Tham, J.Y.: Learning attribute representations with localization for flexible fashion search. In: CVPR. pp. 7708–7717 (2018)
3. Bao, H., Dong, L., Wei, F.: Beit: Bert pre-training of image transformers. arXiv preprint arXiv:2106.08254 (2021)
4. Brown, T.B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al.: Language models are few-shot learners. In: NeurIPS (2020)
5. Cao, J., Li, Y., Zhang, Z.: Partially shared multi-task convolutional neural network with local constraint for face attribute learning. In: CVPR. pp. 4290–4299 (2018)
6. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: ECCV. pp. 213–229 (2020)
7. Chen, H., Gallagher, A., Girod, B.: Describing clothing by semantic attributes. In: ECCV. pp. 609–623 (2012)
8. Chen, T., Saxena, S., Li, L., Fleet, D.J., Hinton, G.: Pix2seq: A language modeling framework for object detection. arXiv preprint arXiv:2109.10852 (2021)
9. Cheng, X., Lin, H., Wu, X., Yang, F., Shen, D., Wang, Z., Shi, N., Liu, H.: Mltr: Multi-label classification with transformer. arXiv preprint arXiv:2106.06195 (2021)
10. Cubuk, E.D., Zoph, B., Shlens, J., Le, Q.: Randaugment: Practical automated data augmentation with a reduced search space. In: NeurIPS. pp. 18613–18624 (2020)
11. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. In: NAACL (2019)
12. Doersch, C., Gupta, A., Zisserman, A.: Crosstransformers: spatially-aware few-shot transfer. In: NeurIPS (2020)
13. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. In: ICLR (2021)
14. Duan, K., Parikh, D., Crandall, D., Grauman, K.: Discovering localized attributes for fine-grained recognition. In: CVPR. pp. 3474–3481 (2012)
15. Farhadi, A., Endres, I., Hoiem, D., Forsyth, D.: Describing objects by their attributes. In: CVPR. pp. 1778–1785 (2009)
16. Feris, R.S., Lampert, C., Parikh, D.: Visual Attributes. Springer (2017)
17. Guo, H., Zheng, K., Fan, X., Yu, H., Wang, S.: Visual attention consistency under image transforms for multi-label image classification. In: CVPR. pp. 729–739 (2019)
18. Hand, E.M., Chellappa, R.: Attributes for improved attributes: A multi-task network utilizing implicit and explicit relationships for facial attribute classification. In: AAAI (2017)
19. He, K., Xinlei, C., Xie, S., Li, Y., Dollár, P., Girshick, R.: Masked autoencoders are scalable vision learners. arXiv preprint arXiv:2106.08254 (2021)
20. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR. pp. 770–778 (2016)
21. He, K., Fu, Y., Zhang, W., Wang, C., Jiang, Y.G., Huang, F., Xue, X.: Harnessing synthesized abstraction images to improve facial attribute recognition. In: IJCAI. pp. 733–740 (2018)
22. Huang, S., Li, X., Cheng, Z.Q., Zhang, Z., Hauptmann, A.: Gnas: A greedy neural architecture search method for multi-attribute learning. In: ACM MM. pp. 2049–2057 (2018)

23. Jia, J., Chen, X., Huang, K.: Spatial and semantic consistency regularizations for pedestrian attribute recognition. In: ICCV. pp. 962–971 (2021)
24. Kalayeh, M.M., Gong, B., Shah, M.: Improving facial attribute prediction using semantic segmentation. In: CVPR. pp. 6942–6950 (2017)
25. Lanchantin, J., Wang, T., Ordonez, V., Qi, Y.: General multi-label image classification with transformers. In: CVPR. pp. 16478–16488 (2021)
26. Li, D., Chen, X., Huang, K.: Multi-attribute learning for pedestrian attribute recognition in surveillance scenarios. In: ACPR. pp. 111–115 (2015)
27. Li, D., Chen, X., Zhang, Z., Huang, K.: Pose guided deep model for pedestrian attribute recognition in surveillance scenarios. In: ICME. pp. 1–6 (2018)
28. Li, D., Zhang, Z., Chen, X., Huang, K.: A richly annotated pedestrian dataset for person retrieval in real surveillance scenarios. TIP **28**(4), 1575–1590 (2018)
29. Li, J., Zhao, F., Feng, J., Roy, S., Yan, S., Sim, T.: Landmark free face attribute prediction. TIP **27**(9), 4651–4662 (2018)
30. Li, W., Duan, Y., Lu, J., Feng, J., Zhou, J.: Graph-based social relation reasoning. In: ECCV. pp. 18–34 (2020)
31. Li, W., Huang, X., Lu, J., Feng, J., Zhou, J.: Learning probabilistic ordinal embeddings for uncertainty-aware regression. In: CVPR. pp. 13896–13905 (2021)
32. Li, W., Lu, J., Feng, J., Xu, C., Zhou, J., Tian, Q.: Bridgenet: A continuity-aware probabilistic network for age estimation. In: CVPR. pp. 1145–1154 (2019)
33. Liu, P., Liu, X., Yan, J., Shao, J.: Localization guided learning for pedestrian attribute recognition. In: BMVC (2018)
34. Liu, S., Zhang, L., Yang, X., Su, H., Zhu, J.: Query2label: A simple transformer way to multi-label classification. arXiv preprint arXiv:2107.10834 (2021)
35. Liu, X., Zhao, H., Tian, M., Sheng, L., Shao, J., Yi, S., Yan, J., Wang, X.: Hydraplus-net: Attentive deep features for pedestrian analysis. In: ICCV. pp. 350–359 (2017)
36. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. In: ICCV (2021)
37. Liu, Z., Luo, P., Qiu, S., Wang, X., Tang, X.: Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In: CVPR. pp. 1096–1104 (2016)
38. Liu, Z., Luo, P., Wang, X., Tang, X.: Deep learning face attributes in the wild. In: ICCV. pp. 3730–3738 (2015)
39. Mao, L., Yan, Y., Xue, J.H., Wang, H.: Deep multi-task multi-label cnn for effective facial attribute classification. TAC (2020)
40. Meng, Z., Adluru, N., Kim, H.J., Fung, G., Singh, V.: Efficient relative attribute learning using graph neural networks. In: ECCV. pp. 552–567 (2018)
41. Nguyen, H.D., Vu, X.S., Le, D.T.: Modular graph transformer networks for multi-label image classification. In: AAAI. pp. 9092–9100 (2021)
42. Perrett, T., Masullo, A., Burghardt, T., Mirmehdi, M., Damen, D.: Temporal-relational crosstransformers for few-shot action recognition. In: CVPR. pp. 475–484 (2021)
43. Peters, M.E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., Zettlemoyer, L.: Deep contextualized word representations. In: NAACL (2018)
44. Radford, A., Narasimhan, K., Salimans, T., Sutskever, I.: Improving language understanding by generative pre-training (2018)
45. Rudd, E.M., Günther, M., Boult, T.E.: Moon: A mixed objective optimization network for the recognition of facial attributes. In: ECCV. pp. 19–35 (2016)
46. Sarafianos, N., Xu, X., Kakadiaris, I.A.: Deep imbalanced attribute classification using visual attention aggregation. In: ECCV. pp. 680–697 (2018)

47. Sarfraz, M.S., Schumann, A., Wang, Y., Stiefelhagen, R.: Deep view-sensitive pedestrian attribute inference in an end-to-end model. In: BMVC (2017)
48. Shao, J., Kang, K., Change Loy, C., Wang, X.: Deeply learned attributes for crowded scene understanding. In: CVPR. pp. 4657–4666 (2015)
49. Shin, M.: Semi-supervised learning with a teacher-student network for generalized attribute prediction. In: ECCV. pp. 509–525 (2020)
50. Shu, Y., Yan, Y., Chen, S., Xue, J.H., Shen, C., Wang, H.: Learning spatial-semantic relationship for facial attribute recognition with limited labeled data. In: CVPR. pp. 11916–11925 (2021)
51. Tang, C., Sheng, L., Zhang, Z., Hu, X.: Improving pedestrian attribute recognition with weakly-supervised multi-scale attribute-specific localization. In: ICCV. pp. 4997–5006 (2019)
52. Taylor, W.L.: "cloze procedure": A new tool for measuring readability. Journalism Quarterly **30**(4), 415–433 (1953)
53. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: NeurIPS. pp. 5998–6008 (2017)
54. Wang, Y., Xu, Z., Wang, X., Shen, C., Cheng, B., Shen, H., Xia, H.: End-to-end video instance segmentation with transformers. In: CVPR. pp. 8741–8750 (2021)
55. Wang, Z.J., Turko, R., Chau, D.H.: Dodrio: Exploring transformer models with interactive visualization. In: ACL (2021)
56. Yu, B., Li, W., Li, X., Lu, J., Zhou, J.: Frequency-aware spatiotemporal transformers for video inpainting detection. In: ICCV. pp. 8188–8197 (2021)
57. Zhang, C., Gupta, A., Zisserman, A.: Temporal query networks for fine-grained video understanding. In: CVPR. pp. 4486–4496 (2021)
58. Zhang, H., Cisse, M., Dauphin, Y.N., Lopez-Paz, D.: mixup: Beyond empirical risk minimization. In: ICLR (2018)
59. Zhang, Y., Zhang, P., Yuan, C., Wang, Z.: Texture and shape biased two-stream networks for clothing classification and attribute recognition. In: CVPR. pp. 13538–13547 (2020)
60. Zhao, X., Yang, Y., Zhou, F., Tan, X., Yuan, Y., Bao, Y., Wu, Y.: Recognizing part attributes with insufficient data. In: ICCV. pp. 350–360 (2019)
61. Zheng, S., Lu, J., Zhao, H., Zhu, X., Luo, Z., Wang, Y., Fu, Y., Feng, J., Xiang, T., Torr, P.H., et al.: Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In: CVPR. pp. 6881–6890 (2021)