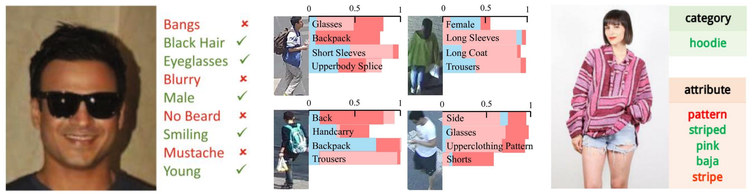
Label2Label: A Language Modeling Framework for Multi-Attribute Learning

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

Department of Automation, Tsinghua University

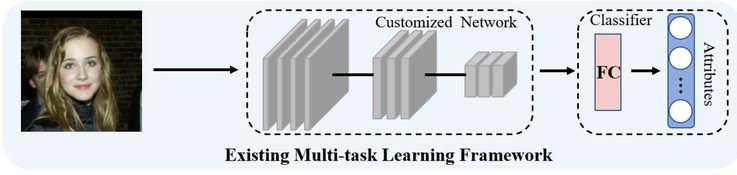
Personal Homepage: https://li-wanhua.github.io Project Website: https://li-wanhua.github.io/Label2Label/ Code: https://github.com/Li-Wanhua/Label2Label Video: https://www.youtube.com/watch?v=999znKklDb4

1. INTRODUCTION



Facial attribute recognition | Pedestrian attribute recognition | Cloth attribute prediction

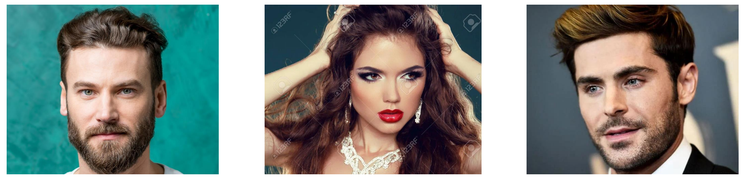
- Attributes are mid-level semantic concepts for objects which are shared across categories. We can describe objects with a wide variety of attributes.
- Multi-attribute learning aims to predict the attributes of objects accurately. It involves many important tasks including facial attribute recognition, pedestrian attribute prediction, and cloth attribute recognition.



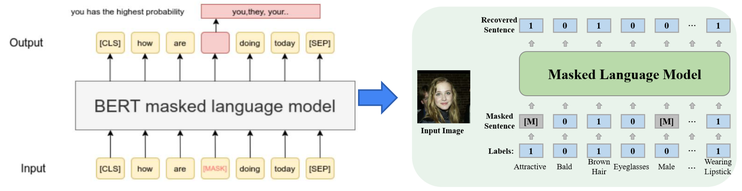
Existing Multi-task Learning Framework

- Most existing approaches adopt a multi-task learning framework, which formulates multi-attribute recognition as a multi-label classification task and simultaneously earns multiple binary classifiers.

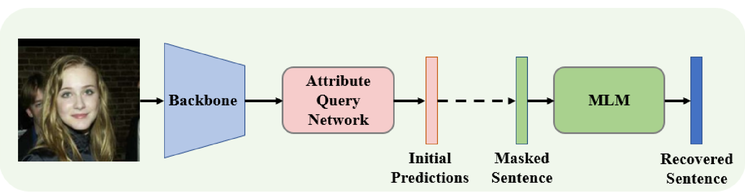
2. LANGUAGE MODELING FRAMEWORK



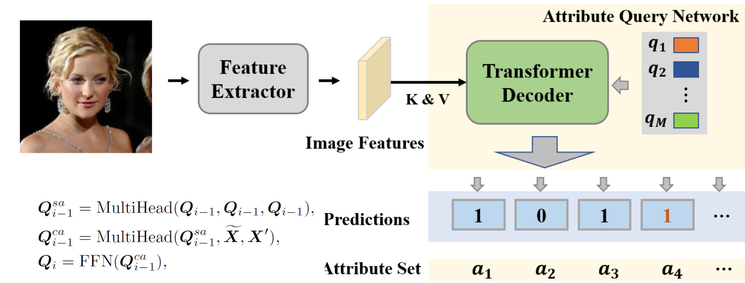
- For a given sample, many of its attributes are correlated.
- Modeling the complex relations among different attributes is the key challenge.



- As a representative work, BERT utilizes a masked language model to capture the word co-occurrence and language structure.
- We regard each attribute label as a "word", which describes the current state of the sample from a certain point of view.
- As each sample is attached with multiple attribute labels which are used to depict the same object, these "words" can be organized as an unordered yet meaningful "sentence".
- Then we use masked language model to learn attribute relations.



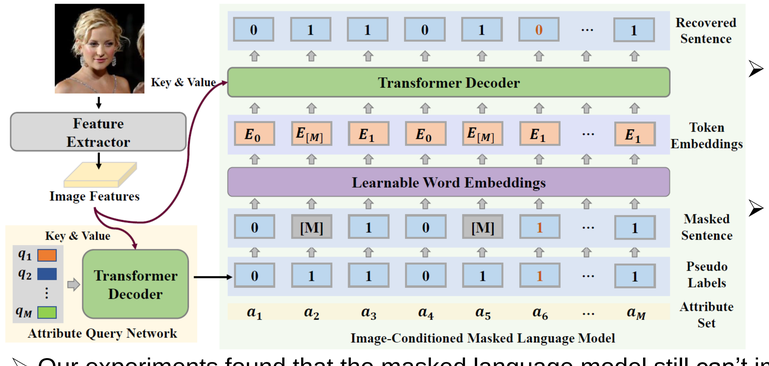
- We cannot access these labels for inference. We introduce an attribute query network to generate the initial attribute predictions.
- These predictions are then treated as pseudo-labels and used as input to the masked language model.



$$Q_{i-1}^{sa} = \text{MultiHead}(Q_{i-1}, Q_{i-1}, Q_{i-1}),$$
$$Q_{i-1}^{ca} = \text{MultiHead}(Q_{i-1}^{sa}, \widetilde{X}, X'),$$
$$Q_i = \text{FFN}(Q_{i-1}^{ca}),$$

- Our attribute query network learns a set of query vectors. Each query corresponds to an attribute type.
- Each query vector pools the attribute-related features from the image features with Transformer decoder layers and generates the corresponding response vector.
- We learn a binary classifier for each response vector to generate the initial attribute predictions.

3. IMAGE-CONDITIONED MASKED LANGUAGE MODEL



- Masked Language Model:
  $P(y_1, y_2, ..., y_M | \mathcal{M}(y_1), \mathcal{M}(y_2), ..., \mathcal{M}(y_M))$
  - Statistical Attribute Correlations
  - Two-stage Training
- Image-Conditioned MLM:
  $P(y_1, y_2, ..., y_M | \boldsymbol{x}, \mathcal{M}(y_1), \mathcal{M}(y_2), ..., \mathcal{M}(y_M))$
  - Instance-wise Attribute Relations
  - End-to-end Training

- Our experiments found that the masked language model still can't improve the performance.
- The first problem is that masked language model only captures statistical attribute correlations. The second problem is that masked language model and attribute query network cannot be jointly trained.
- To address these issues, we propose an image-conditioned masked language model, which recovers the masked "words" conditioned on the masked "sentence" and image features.
- With the conditions of the image, the constructed task is an accurate one-to-one mapping.
- Image-conditioned masked language model and the attribute query network can use shared image features, which enables them to be jointly optimized with a one-stage framework.

4. EXPERIMENTAL RESULTS

- Extensive experimental results show that Image-Conditioned MLM is the key to success.

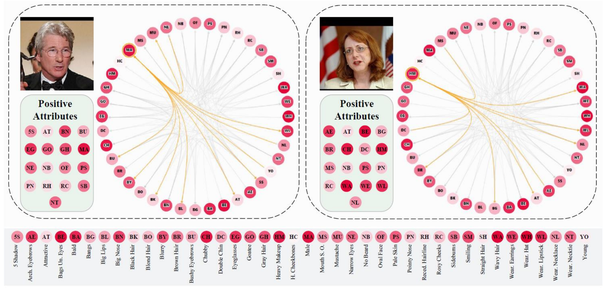
Table 5. Ablation experiments with different backbones.

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 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**

- Our method also generates better interpretable predictions. We visualize the attention scores in the Transformer decoder. we see that the related attributes tend to have higher attention scores.



- Our method attains competitive results on several multi-attribute learning tasks.

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

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**