## Evaluating Attribute Comprehension in Large Vision-Language Models

Haiwen Zhang<sup>1</sup>, Zixi Yang<sup>1</sup>, Yuanzhi Liu<sup>1</sup>, Xinran Wang<sup>1</sup>, Zheqi He<sup>2</sup>, Kongming Liang\*<sup>1</sup>, and Zhanyu Ma<sup>1</sup>

- <sup>1</sup> Beijing University of Posts and Telecommunications, Beijing, China liangkongming@bupt.edu.cn
  - <sup>2</sup> Beijing Academy of Artificial Intelligence, Beijing, China

**Abstract.** Currently, large vision-language models have gained promising progress on many downstream tasks. However, they still suffer many challenges in fine-grained visual understanding tasks, such as object attribute comprehension. Besides, there have been growing efforts on the evaluations of large vision-language models, but lack of in-depth study of attribute comprehension and the visual language fine-tuning process. In this paper, we propose to evaluate the attribute comprehension ability of large vision-language models from two perspectives: attribute recognition and attribute hierarchy understanding. We evaluate three visionlanguage interactions, including visual question answering, image-text matching, and image-text cosine similarity. Furthermore, we explore the factors affecting attribute comprehension during fine-tuning. Through a series of quantitative and qualitative experiments, we introduce three main findings: (1) Large vision-language models possess good attribute recognition ability, but their hierarchical understanding ability is relatively limited. (2) Compared to ITC, ITM exhibits superior capability in capturing finer details, making it more suitable for attribute understanding tasks. (3) The attribute information in the captions used for fine-tuning plays a crucial role in attribute understanding. We hope this work can help guide future progress in fine-grained visual understanding of large vision-language models. The code will be available at Attribute-Comprehension-of-VLMs.

**Keywords:** Large Vision-Language Models  $\cdot$  Attribute Recognition  $\cdot$  Hierarchical Understanding.

## 1 Introduction

Visual attributes [14,19,18,2,20,28] are important components of objects, enabling models to describe object information more accurately, which benefits various downstream tasks, including compositional reasoning [5] and visual question answering [29,6]. In recent years, large vision-language models have achieved remarkable achievements, while some works have identified many problems in large vision-language models, such as hallucination issues [13] and social bias [4]. A direct question is: How do large-scale vision-language models perform in attribute

understanding? Previous works [27,26] find that the model cannot correctly establish the connection between objects and attributes, resulting in degraded visual question answering (VQA) performance. [2,30] indicate that compared to zero-shot image classification, the absolute performance of attribute recognition is surprisingly low. However, previous studies simply evaluate the large vision-language models on a limited attribute set instead of large-scale attributes in the wild scenario. Besides, understanding attributes not only demands accurate recognition of individual attributes but also comprehending the semantic relationships between attributes, such as hierarchical relationships.

Therefore, in this work, we utilize the VAW dataset [19] to enlarge the scale of wild visual attributes and supplement the hierarchical annotations to evaluate the understanding between attribute semantic relationships. The performance of mainstream large vision-language models in understanding large-scale wild attributes is evaluated from two perspectives: attribute recognition and attribute hierarchical relationship understanding. Attribute recognition requires the model to predict the object attributes correctly. While attribute hierarchical relationship understanding requires the model to predict the existence of the parent attribute when it predicts the existence of a child attribute. Similarly, when predicting the absence of a parent attribute, it should also predict the absence of its child attributes. As illustrated in Fig.1, the model predicts the presence of the positive attribute 'navy blue', but it outputs 'No' for its parent attribute 'blue', thus violating the hierarchy. We conduct evaluation through three approaches: visual question answering (VQA), image-text matching (ITM), and image-text cosine similarity (ITC), reporting attribute understanding ability of different models from two aspects and comparing the distinctions in evaluation methodologies. Experiments reveal that current large vision-language models have achieved a certain level of attribute comprehension. BLIP [11], BLIP2 [10], ALBEF [12], and mPLUG [9] even surpass supervised models in recognizing tail attributes.

Secondly, we examine why large vision-language models achieve comparable levels of attribute comprehension capability, even though fine-tuned on coarse image-text pairs rather than high-quality attribute annotations such as VAW [19]. This raises questions about which factors influence the understanding of attributes during fine-tuning. To study the question, we further investigate two aspects: image resolution and the diversity of attribute information in the captions. Our experimental results indicate that the attribute information within the captions plays an important role in the process while image resolution appears to be less important for attribute comprehension.

Overall, our contributions can be summarized as follows: (1). We propose an evaluation framework for assessing the attribute understanding capabilities of large vision-language models in terms of attribute recognition and hierarchical relationship understanding. (2). Through experiments, we find that ITM possesses better detail-capturing ability than ITC and that the two methods respond differently to different templates. ITM requires more detailed templates while ITC performs better when provided only attributes in the templates. (3).

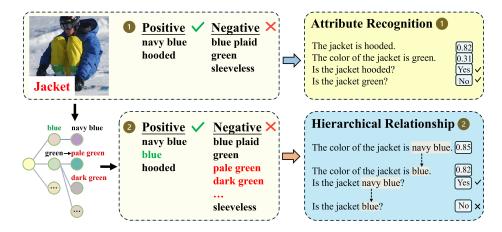


Fig. 1: Overview of the evaluation process. Original annotations of VAW are used for attribute recognition, while complementary annotations are used to evaluate hierarchical relationship understanding. We utilize the attribute tree [14] to perform complementation, and the inference results of ITM and ITC are presented as scores. For VQA, the results are presented as either 'Yes' or 'No'.

We examine the impact of image resolution as well as attribute information in captions used for fine-tuning and offer insights into the construction of future image-text pairs.

## 2 Attribute Understanding Benchmark

This section presents the evaluation process of attribute recognition and hierarchical relationship understanding. We introduce three evaluation methods: image-text cosine similarity, image-text matching and visual question answering. All evaluations are conducted on the VAW test set [19], which encompasses 620 attributes and covers various attribute types, including color, material, shape, size, texture, action, state, and others.

## 2.1 Formulation

Given an image x of an object o, x is partially labeled by its corresponding attribute vector  $y = \{y_a\}_{a=1}^A$ , where A denotes the total number of attributes and  $y_a \in \{-1,0,1\}$  denotes whether the attribute a is present in the image ('1'), absent ('-1') or unknown ('0'). The positive, negative, and un-annotated attribute sets of x are denoted as  $\mathcal{P}_x = \{a|y_a = 1\}$ ,  $\mathcal{N}_x = \{a|y_a = -1\}$ , and  $\mathcal{U}_x = \{a|y_a = 0\}$ , respectively. A model with good attribute recognition capability must correctly predict whether an attribute is present in the input image.

There are also hierarchical relationships between attributes, which can be considered as a directed acyclic graph (DAG) denoted as  $\mathcal{T}$  [14]. The adjacency

#### 4 F. Author et al.

matrix of  $\mathcal{T}$  is denoted as  $\mathcal{R}^*$ , where  $\mathcal{R}^*_{ij} = 1$  if and only if attribute i is the parent of attribute j. We denote the descendant set of i as  $\mathcal{D}_i$ , if there is a path from j to i in  $\mathcal{T}$ , we say j is a descendant of i, formulated as  $j \in \mathcal{D}_i$ . A model understanding attribute hierarchical relationship must predict an attribute's all ancestors present when it predicts the attribute present, and an attribute's all descendants absent when it predicts the attribute absent.

### 2.2 Evaluation Aspects

Attribute Recognition We show the evaluation process in Fig.1. To perform inference, we first populate the template with the attribute and object names to obtain the prompt. Second, the obtained prompt and the image are fed into the vision-language model to get the inference results. This process is repeated three times to perform different evaluation approaches, including ITC, ITM, and VQA. Through ITM and ITC, we can get the inference scores. And it is anticipated that the scores of positive attributes are higher than those of negative attributes. Through VQA, we obtain the answers of 'Yes' and 'No'. And it is expected that the predicted answers for positive attributes are 'Yes' and 'No' for negative attributes. There are a total of 224,855 positive and negative attributes for 31,819 instances in the test set of VAW. For attribute recognition, we use accuracy to report the metrics for VQA and mean average precision (mAP) to report the metrics for ITM and ITC.

Hierarchical Relationship Understanding To evaluate the understanding of hierarchical relationships, we first utilize the attribute hierarchy [14] to complement the annotations. A complementary example is presented in Fig.1. For the positive attribute 'navy blue', we add its parent attribute 'blue' to the positive attribute annotation. Since 'hooded' has no parent attribute, there is no need to perform a supplement. For the negative attribute 'green', we incorporate its descendant attributes into the negative attribute annotation, including 'dark green', 'pale green', etc. Similarly, we do not need to operate for 'blue plaid' and 'sleeveless' since they do not have descendant attributes. After complementation, we have a total of 396,242 positive and negative attributes. Subsequently, the inference operation is conducted on the complementary dataset. We rely on the constraint violation (CV) and mean average precision post coherence correction (CmAP) metrics to report the models' performance in understanding hierarchical relationships. It is hoped that the scores for parent attributes are higher than those of child attributes. In Fig.1, the score of 'navy blue' is higher than 'blue', and the model outputs 'No' for 'blue' while it predicts the presence of 'navy blue', which violates hierarchy.

### 2.3 Evaluation Methodologies

Image-text Cosine Similarity (ITC) Many large vision-language models [12,22,11,10,9] use image-text cosine similarity as one of the pre-training objectives to align image and text representations in the joint embedding space. Given

the image embedding I of x and the text embedding T of attribute prompt  $t_a$ : 'The {attribute type} of the {object} is {attribute}.', we can get the probability of attribute a present in x:

$$p_a = \sigma(\cos(I, T)),\tag{1}$$

where  $cos(\cdot)$  is the cosine similarity function and  $\sigma$  is the sigmoid function to scale the logits into [0,1]. This template is shared by ITM and ITC, while we also explore other templates in the ablation study.

Image-text Matching (ITM) It is also a training objective shared by many models, including [12,11,10,9], which predicts whether a pair of image and text is matched or mismatched. During the training process, hard negative samples from ITC are selected for better fusion between visual and textual information. The ITM head outputs an array with two scores, the first is the mismatch score, and the other is the match score. The evaluation process of ITM is similar to ITC, however, embeddings will be entered into the ITM head to get scores:

$$S = \sigma(F_{itm}(E_{mm}(I,T))), \tag{2}$$

where  $E_{mm}$  is a multimodal encoder,  $F_{itm}$  is the ITM head, and  $\sigma$  is the softmax function. Then  $p_a$ , the probability of attribute a present in x can be obtained by taking the second element of S.

Visual Question Answering (VQA) Visual question answering requires the model to answer the question according to the corresponding image. The VQA models used in the experiment are fine-tuned on the VQA dataset [6], except for MiniGPT-4 [31], which only provides the pre-trained model. For fairness, we limit the candidate answers to containing only two answers: 'Yes' and 'No'. The question template is 'Is the {object} {attribute}?', while for material attributes it is 'Is the {object} made of {attribute}?'. Since BLIP2 [10] does not provide an interface to select from candidate answers, we do not perform evaluations on it.

#### 2.4 Evaluation Metrics

To evaluate the attribute understanding capability of different models, we utilize four evaluation metrics.

- (1) Accuracy: It serves as an assessment of the overall correctness of a model's responses to questions.
- (2) Mean average precision (mAP): Mean average precision is a prominent metric particularly applied to attribute prediction and multi-label classification. Due to the partial attributes annotation of VAW, we only consider annotated attributes.
- (3) Mean average precision post coherence correction (CmAP) [17]: It post-processes attribute prediction scores to enforce hierarchical constraints

within the model's outputs. Specifically, for a non-leaf attribute, it computes the maximum prediction score among its descendant attributes. Large deviations between mAP and CmAP indicate poorer hierarchical understanding. Given a non-leaf attribute i, its post-processing probability:

$$p_i^* = \max\{p_i, \max\{p_i | j \in D_i\}\}.$$
(3)

(4) Constraint violation (CV) [17] is a metric utilized to assess the hierarchical constraint violations present in a model's predictions. It measures the extent to which model predictions deviate from hierarchical constraints. For ITM and ITC, we compute the CV based on the models' scores

$$CV_{ITM/ITC} = \frac{1}{|\mathcal{K}||\mathcal{T}|} \sum_{k=1}^{|\mathcal{K}|} \sum_{j \in D_i} 1\left(p_i^{(k)} - p_j^{(k)}\right) < 0, \tag{4}$$

where  $|\mathcal{K}|$  is the number of test images and  $|\mathcal{T}|$  is the number of edges in the attribute tree  $\mathcal{T}$ . For VQA, we calculate the CV based on the models' answers. We use  $T^*$  to represent the number of edges present in the image after complementation in the manner described in 2.2. While  $F^*$  is to represent the number of edges that violate hierarchy, then we have:

$$CV_{VQA} = \frac{1}{|\mathcal{K}|} \sum_{k=1}^{|\mathcal{K}|} \frac{F^*}{T^*}.$$
 (5)

## 2.5 Comparison Methods

The models compared in our experiments are as follows. We list the corresponding model weights of evaluated large vision-language models in Table 1.

ResNet-Baseline [18] provides a baseline for object attribute recognition. VAW [19] proposes a strong baseline model along with supervised contrastive learning using negative-label expansion. CLIP [22] is trained on 400 million image-text pairs, using a text encoder and a visual encoder to map images and texts to the same feature space. ALBEF [12] employs contrastive alignment of image and text representations before fusion, enabling robust vision-language learning without needing bounding box annotations or high-resolution images. BLIP [11] adeptly handles understanding and generation tasks, leveraging noisy web data through caption bootstrapping. BLIP2 [10] bridges frozen pre-trained image encoders and language models by training a lightweight querying transformer. mPLUG [9] proposes a cross-modal skipped-connected network to address the problem of information asymmetry between image and text and improve computational efficiency. MiniGPT-4 [31] aligns a frozen text encoder and a frozen LLM using one projection layer to achieve advanced multi-modal capabilities.

Table 1: This table represents the weight parameters and corresponding datasets for our evaluated models. A: Conceptual Captions [25]; B: SBU Captions [16]; C: COCO [15]; D: Visual Genome [8]; E: Conceptual 12M [3]; F: VQA [1]; G: LAION400M [23]; H: Flickr30K [21].

Models	Pre-training	Weight	Fine-tuning	VQA	ITM	ITC
Wodels	datasets		datasets			
CLIP [22]	WebImageText	ViT-B/32(151.3M)	_			<b>√</b>
ALBEF[12]	ABC	albef_vqav2_lavis(580.7M)	F	<b>√</b>		
ALDEF [12]	DE	albef_coco_retrieval_lavis(419.1M)	C		✓	✓
BLIP [11]	BCD	model_base_vqa_capfilt_large(361.5M)	F	<b>√</b>		
DLII [11]	ΕG	model_base_retrieval_coco(223.7M)	C		✓	✓
BLIP2 [10]	ABCDEG	blip2_finetune_coco(1173.2M)	C		<b>√</b>	<b>√</b>
mPLUG [9]	ABC	mplug_visual-question-answering_coco_large_en(1494.3M)	F	✓		
III LUG [9]	DE	mplug_image-text-retrieval_flickr30k_large_en(1217.8M)	H		✓	✓
MiniGPT-4 [31]	ABEG	pretrained_minigpt4_7b(7832.6M)	_	<b>√</b>		

Table 2: Comparison of attribute recognition ability with the close-set models on the VAW dataset [19]. The best overall results are in bold. The best results in close-set models, ITC, and ITM are highlighted in Green, Red, and Blue, respectively.

Methods	Overall	Class	imbalanc	e(mAP)			Attri	bute 1	types (m	AP)		
Wethous	(mAP)	Head	Medium	Tail	Color	Material	Shape	Size	Texture	Action	State	Others
Close-set models												
ResNet-Baseline [18]	63.0	71.1	59.4	43.0	58.5	66.3	65.0	64.5	63.1	53.1	_	66.7
VAW [19]	68.3	76.5	64.8	48.0	70.4	75.6	68.3	69.4	68.4	60.7	_	69.5
Open-set models ima	ge-text o	cosine	similarity									
CLIP [22]	42.1	43.6	43.2	33.2	33.0	33.6	36.8	42.3	43.3	38.2	41.7	47.1
ALBEF [12]	49.0	48.5	51.8	42.5	38.2	37.9	42.8	43.5	44.7	54.6	46.6	54.2
BLIP [11]	52.2	52.5	54.4	44.5	44.2	44.6	44.9	45.5	48.6	53.2	49.2	57.2
BLIP2 [10]	58.0	56.4	60.9	55.3	49.9	52.9	53.1	47.0	55.4	63.9	53.1	62.0
mPLUG [9]	45.3	45.0	47.4	40.4	35.5	34.0	41.0	41.4	39.3	45.5	46.01	50.7
Open-set models ima	ge-text 1	natch	ing									
ALBEF [12]	56.3	55.9	58.9	49.7	51.6	47.5	47.6	45.0	54.8	59.3	51.4	60.7
BLIP [11]	59.1	57.9	62.1	54.3	57.5	51.7	49.0	47.2	58.1	57.1	52.5	63.7
BLIP2 [10]	62.7	60.5	65.7	61.5	57.2	56.1	57.5	46.2	59.4	63.2	55.2	67.9
mPLUG [9]	60.0	58.8	62.9	55.6	56.0	53.6	49.7	45.8	54.7	61.2	58.5	64.3

## 3 Experimental Results

In this section, we show the experimental results of our proposed benchmark and compare different models through a series of analyses.

## 3.1 Comparison with Close-Set Models

We demonstrate the attribute understanding ability of different models in terms of attribute recognition and hierarchical relationship understanding, each of which is evaluated in three ways: VQA, ITM, and ITC. For attribute recognition results, Table 2 presents a comparison between close-set models and the open-set models using ITC and ITM. We find that large vision-language models exhibit substantial attribute comprehension capabilities. For instance, the

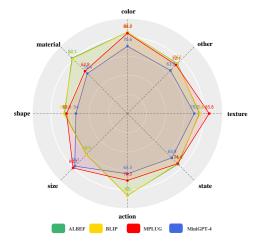


Fig. 2: Attribute recognition accuracy of different models. The performance is assessed across eight attribute types within the VAW dataset.

Table 3: Experimental results of hierarchical understanding ability of different models. The computation of CV for the close-set models is consistent with that for ITM and ITC, so we do not compare it with VQA. The best results are in bold.

Method		$CV \downarrow$		CmAP ↑							
Close-set models											
VAW [19]	_	23	.7	68	.3						
Open-set models	VQA	ITM	ITC	ITM	ITC						
CLIP [22]	_	_	58.2	_	47.9						
ALBEF [12]	8.24	43.1	45.0	60.7	54.3						
BLIP [11]	8.75	34.7	34.1	63.3	57.2						
BLIP2 [10]	_	41.5	42.7	66.7	62.5						
mPLUG [9]	9.84	48.15	57.4	64.1	50.9						
MiniGPT-4 [31]	11.37	_	_	-	_						

overall mAP of BLIP2's ITM is very close to the ResNet-Baseline, which is trained on the VAW dataset. Additionally, for tail attributes, the performance of mPLUG, BLIP, and BLIP2 using ITM is even much better than supervised models, indicating that the data imbalance of the training set impairs the performance of supervised models. In contrast, large vision-language models with strong generalization capabilities can mitigate this issue.

Fig.2 represents the attribute recognition results of VQA. Among them, AL-BEF, BLIP, and mPLUG obtain comparable recognition results, but mPLUG and the previous two models have differences in recognizing different attribute types. Table 3 shows the results of hierarchical relationship understanding capability, we find large vision-language models can comprehend certain hierarchical relationships. However, there are some deviations between CmAP and mAP. Besides, the CV metrics for ITM and ITC also have a significant gap compared to the supervised model, which demonstrates less favorable hierarchical understanding capability since the scores of some parent attributes are lower than those of their descendent attributes. The CV metrics for VQA seem to be better since it only predicts the presence and absence of an attribute without the limitation of scores. In both attribute recognition and hierarchical relationship understanding evaluations, BLIP2 achieves the best ITM and ITC inference results.

#### 3.2 Analysis

What Affects Hierarchical Relationship Understanding. According to Table 3, mPLUG understands hierarchical relationships better than MiniGPT-



Fig. 3: Hierarchical relationship understanding comparison between mPLUG and MiniGPT-4, where the parent attributes are highlighted in red and the children attributes are highlighted in green.

4. Through the visualization results in Fig.3, we identify two factors that may influence hierarchical relationship understanding. First, other objects in the image contain the attribute. Like the 'olive green planter' in Fig.3, though both models can correctly determine that green is a negative attribute of the planter, MiniGPT-4 mistakenly identifies it as olive green, which violates hierarchy. The question is about the planter rather than the plant, but it fails to locate the right area when recognizing 'olive green'. Second, the attribute phrase is not present on the object, but the attribute contained in the phrase is present on the object. Such as the 'wearing white dog' in Fig.3, both models misidentify when recognizing whether the dog is wearing white or not, while they can distinguish that the dog is not dressed. It is probably because the dog is white and they cannot understand 'wearing white'. These visualizations suggest that sometimes models behave like 'bags-of-words' [27] and lack accurate semantic comprehension, thus impairing attribute understanding.

Effects of Different Evaluation Methods. In Table 2, we observe that the mAP values of ITM are significantly higher than those of ITC for the same model. This phenomenon can be explained by analyzing the score distribution. Here, we refer to [7] to perform analysis. Fig.4 Right shows that ITM has a distinct threshold between positive and negative attributes, with most negative attribute scores concentrated in the low-score region. In contrast, ITC lacks a clear boundary between positive and negative attributes, leading to a lower

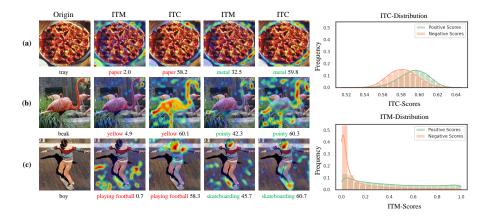


Fig. 4: **Left**: Comparison between Grad-CAM[24] visualizations for BLIP ITM and ITC, corresponding to the prompt. Scores are displayed below the picture. Prompts: (a). Negative: The material of the tray is **paper**. Positive: The material of the tray is **metal**. (b). Negative: The color of the beak is **yellow**. Positive: The shape of the beak is **pointy**. (c). Negative: The sport activity of the boy is **playing football**. Positive: The sport activity of the boy is **skateboarding**. Scores are present below the picture. **Right**: The distribution of positive and negative attribute prediction scores for BLIP [11].

mAP. To further analyze the results, we use Grad-CAM [24] to visualize the interpretable components that contribute to the prediction and the degree of contribution of each feature. The visualizations in Fig.4 Left show that both ITM and ITC can correctly locate the target object when provided with a prompt of a positive attribute. However, when recognizing negative attributes that are not present on the object, ITC only concentrates on the entire object while ITM could explore relevant regions of the object that may be associated with that attribute.

As shown in Fig.4 left (c), ITC still focuses on the boy when provided with the negative attribute 'playing football', whereas ITM attempts to explore the specified attribute and successfully directs attention to the boy's feet. Additionally, we find that the scores for positive and negative attributes vary widely for ITM, but the difference is not obvious for ITC. For example, in Fig.4 left (c), the difference between positive and negative scores for ITM is 45, while it becomes 2.4 for ITC. The score disparity suggests that ITM is more sensitive to attribute information. In contrast, ITC is limited to object-level recognition and lacks the ability for finer-grained attribute analysis. This may be due to the hard negative sampling strategy in BLIP's ITM training [12,11,9], which enables ITM to capture more nuanced information and achieve better performance in attribute-level analysis.

Table 4: The mAP results for different prompt templates on the VAW dataset. **T1**: 'The object is {attribute}.'; **T2**: 'The {object} is {attribute}.'. We highlight the best template for ITC in blue and for ITM in red.

Tomplete	Object	CLIP [22]	BLIP [11]	mPLUG [9]	ALBEF [12]
Template	Object	ITC	ITC ITM	ITC ITM	ITC ITM
T1		45.7	53.8 54.1	46.9 57.9	51.4 49.3
T2	✓	43.2	53.9 61.2	46.2 61.1	50.1 58.2

## 4 Ablation Study

The evaluation results indicate that the fine-tuned vision-language models possess certain attribute comprehension capabilities. To further explore the factors affecting the fine-tuning performance and the impact of different prompt templates, we conduct ablation experiments.

## 4.1 Effects of Different Templates

In previous ITC and ITM experiments, we use the template that includes attribute types, object names, and attribute names. To investigate the impact of different prompting templates, we attempt two additional templates. The first template includes only attributes: 'The object is {attribute}'; the second template includes both object and attribute names: 'The {object} is {attribute}'. As shown in Table 4, ITC and ITM respond differently to various templates. For ITC, providing only attribute information often yields the best attribute recognition performance. However, for ITM, solely providing attributes is not sufficient; it requires the addition of attribute-dependent information, i.e., the object, to achieve better attribute recognition. This indicates that ITM possesses better detail-capturing capabilities, and thus, it performs better when more detailed descriptions are provided. In contrast, ITC lacks compositional understanding, so adding additional object information can introduce interference, leading to reduced performance in recognizing attributes. However, ITM requires more time and computational resources for inference compared to ITC. Therefore, enhancing the effectiveness of contrastive learning remains crucial.

# 4.2 Exploratory Experiments on Factors Affecting the Performance of Fine-tuning

**Image resolution.** We find that during fine-tuning, most models tend to increase image resolution. For instance, ALBEF [12] takes random image crops of resolution  $256 \times 256$  as input during pretraining, while during fine-tuning, it gets boosted to  $384 \times 384$ . Similarly, BLIP [11] grows from  $224 \times 224$  to  $384 \times 384$ , and for the VQA task, the resolution is  $480 \times 480$ . Considering that higher resolution allows models to capture more subtle visual details, which may be beneficial for attribute understanding, we use ALBEF [12] to explore the effect of image resolution on attribute comprehension during fine-tuning.

Table 5: Evaluation results of different image resolution fine-tuning based on ALBEF [12]. For text retrieval (TR) and image retrieval (IR), we report the average of R@1, R@5, and R@10. The best results are in bold and we also highlight the results inferenced with the officially provided model in red.

				v 1				
Imagen	Dag	COCO Retri	eval(Avg(R@X))	Attribute Recognition(mA				
Image	nes	TR	IR	ITC	ITM			
$256 \times$	256	88.6	77.2	48.5	56.5			
384 ×	384	89.6 ( <mark>89.6</mark> )	78.6 (78.5)	49.6 (49.0)	<b>56.6</b> (56.3)			
480 ×	480	89.9	79.0	48.9	55.9			

The results are shown in Table 5. We use the model obtained by fine-tuning on COCO [15] dataset for the retrieval task. Therefore, we report the text retrieval (TR) and image retrieval (IR) results on the COCO dataset, as well as the attribute recognition results on the VAW test set. To maintain consistency, we reproduce it with the same image resolution as in the original paper (384  $\times$  384). Then, we decrease or increase the resolution while keeping other parameters constant during the process. From Table 5, we can see that as the image resolution increases, the performance on the retrieval task becomes better, while the attribute recognition ability initially rises and then falls. These results demonstrate that image resolution is not the main factor contributing to attribute comprehension capabilities since no significant fluctuations are detected when it is increased from  $256 \times 256$  to  $384 \times 384$ , but it is effective for retrieval tasks.

Attribute Density in Captions. Previous work on attribute predictions requires datasets with high-quality attribute annotations, such as VAW [19]. However, the dataset used for finetuning is composed of image-text pairs, which do not provide a detailed description of the attributes in the image. To investigate the effect of captions in the training set on attribute comprehension, we count the overlapping attributes of the COCO training set with the VAW dataset. The VAW dataset has a total of 620 attributes, 605 of which are included in the training set of COCO. After removing the 15 non-overlapping attributes, the mAPs of ALBEF are increased, while for ITC it is  $49.0 \rightarrow 49.2$  and for ITM it is  $56.3 \rightarrow 56.5$ . This preliminary outcome indicates that the diversity of attributes in the training set may influence the ability to understand attributes. To further validate this hypothesis, we conduct fine-tuning experiments on ALBEF by removing attributes of certain types from captions. Two attempts are made, including 'Color Absent' and 'Material Absent', which correspond to deleting color information and material information, respectively.

We utilize 'bert-base-uncased' for the deletion operation. For example, to delete color information from COCO, first, the color attributes of VAW and the captions of the COCO training set are encoded to obtain the input ids for attributes and captions. Secondly, those in the input ids for captions that overlap with the input ids for attributes are removed. Thirdly, the input ids for captions are decoded and the first letter of each resulting caption is capitalized to obtain

Table 6: Comparison between 'color absent', 'material absent', and our reproduced results on attribute recognition. All experiments are conducted on ALBEF [12]. The changes of color and material types are highlighted in blue and green, respectively.

Methods	Overall	Class	imbalance	e(mAP)		Attribute types (mAP)						
Methods	(mAP)	Head	Medium	Tail	Color	Material	Shape	${\rm Size}$	Texture	Action	${\bf State}$	Others
Open-set models image-text cosine similarity												
Color Absent	46.9	47.0	49.6	38.2	32.3	36.5	40.9	42.7	44.4	51.9	45.4	52.8
Material Absent	48.7	48.3	51.4	41.4	38.2	36.0	42.0	43.4	45.9	52.9	47.0	54.0
Reproduced	49.6	49.2	52.3	42.7	39.0	38.6	42.5	43.6	45.5	54.0	46.8	55.1
Open-set models	Open-set models image-text matching											
Color Absent	55.0	54.5	57.7	48.6	47.6	45.7	47.1	44.3	53.2	59.4	49.7	59.9
Material Absent	56.3	55.8	59.1	49.2	51.6	46.5	46.9	44.8	55.7	59.3	51.7	60.8
Reproduced	56.6	56.1	59.4	50.0	51.5	48.2	47.2	45.1	55.5	59.2	51.6	61.2

a new caption. After the process, the original caption 'Two people are riding a red bike down the street.' is transformed into 'Two people are riding a bike down the street.', where 'red' is removed. However, it is not possible to remove all the color and material information from the captions, such as 'reddish', which is not present in the VAW dataset. Furthermore, the plural form of 'fabric', 'fabrics', cannot be removed since it has a different input id. Therefore, there is still color and material information after deletion, but the diversity is decreased.

We compare the results of removing color and material attributes with our reproduced results in Table 6. Significant declines are observed for the AP of color and material type. This evidence supports the hypothesis that during the fine-tuning process, attribute information contained in the captions enhances the model's attribute understanding capabilities, and the diversity of this attribute information plays an important role in the process. It is also notable that the removal of color information results in a notable reduction in other attribute types, such as 'material'. This could be attributed to the material-related modifications that occur during the process of removing color information, which in turn leads to a decrease in metrics.

## 5 Conclusion

This study examines the ability of large vision-language models to understand object attributes from two aspects: attribute recognition and attribute hierarchical relationship understanding. Our experiments highlight the differences between ITM and ITC in terms of attribute understanding and the impact of different prompt templates. Furthermore, we investigate the factors that affect attribute comprehension during the fine-tuning process, excluding factors that have a minor impact, such as image resolution. We also confirm the significance of diversity of attribute information in captions. With the advancement of large vision-language models, it has become easier to generate image-text pair datasets. This may make it possible to specify dimensions to be enriched to obtain more appropriate datasets for customizing models.

#### References

- Antol, S., Agrawal, A., Lu, J., Mitchell, M., Batra, D., Zitnick, C.L., Parikh, D.: Vqa: Visual question answering. In: Proceedings of the IEEE international conference on computer vision. pp. 2425–2433 (2015)
- Bravo, M.A., Mittal, S., Ging, S., Brox, T.: Open-vocabulary attribute detection. In: IEEE CVPR. pp. 7041–7050 (June 2023)
- 3. Changpinyo, S., Sharma, P., Ding, N., Soricut, R.: Conceptual 12m: Pushing webscale image-text pre-training to recognize long-tail visual concepts. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3558–3568 (2021)
- Chuang, C.Y., Varun, J., Li, Y., Torralba, A., Jegelka, S.: Debiasing visionlanguage models via biased prompts. arXiv preprint 2302.00070 (2023)
- Gao, D., Wang, R., Shan, S., Chen, X.: Cric: A vqa dataset for compositional reasoning on vision and commonsense. IEEE TPAMI 45(5), 5561–5578 (2023). https://doi.org/10.1109/TPAMI.2022.3210780
- Goyal, Y., Khot, T., Summers-Stay, D., Batra, D., Parikh, D.: Making the V in VQA matter: Elevating the role of image understanding in Visual Question Answering. In: IEEE CVPR (2017)
- Huang, X., Huang, Y.J., Zhang, Y., Tian, W., Feng, R., Zhang, Y., Xie, Y., Li, Y., Zhang, L.: Open-set image tagging with multi-grained text supervision. arXiv e-prints pp. arXiv-2310 (2023)
- 8. Krishna, R., Zhu, Y., Groth, O., Johnson, J., Hata, K., Kravitz, J., Chen, S., Kalantidis, Y., Li, L.J., Shamma, D.A., et al.: Visual genome: Connecting language and vision using crowdsourced dense image annotations. International journal of computer vision 123, 32–73 (2017)
- Li, C., Xu, H., Tian, J., Wang, W., Yan, M., Bi, B., Ye, J., Chen, H., Xu, G., Cao, Z., et al.: mplug: Effective and efficient vision-language learning by cross-modal skip-connections. arXiv preprint arXiv:2205.12005 (2022)
- Li, J., Li, D., Savarese, S., Hoi, S.: Blip-2: Bootstrapping language-image pretraining with frozen image encoders and large language models. arXiv preprint arXiv:2301.12597 (2023)
- 11. Li, J., Li, D., Xiong, C., Hoi, S.: Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In: ICML (2022)
- 12. Li, J., Selvaraju, R.R., Gotmare, A.D., Joty, S., Xiong, C., Hoi, S.: Align before fuse: Vision and language representation learning with momentum distillation. In: NeurIPS (2021)
- Li, Y., Du, Y., Zhou, K., Wang, J., Zhao, W.X., Wen, J.R.: Evaluating object hallucination in large vision-language models. arXiv preprint arXiv:2305.10355 (2023)
- Liang, K., Wang, X., Zhang, H., Ma, Z., Guo, J.: Hierarchical visual attribute learning in the wild. In: ACM MM. pp. 3415–3423 (2023)
- Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P.,
   Zitnick, C.L.: Microsoft coco: Common objects in context. In: Computer Vision–
   ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12,
   2014, Proceedings, Part V 13. pp. 740–755. Springer (2014)
- 16. Ordonez, V., Kulkarni, G., Berg, T.: Im2text: Describing images using 1 million captioned photographs. Advances in neural information processing systems **24** (2011)
- Patel, D., Dangati, P., Lee, J.Y., Boratko, M., McCallum, A.: Modeling label space interactions in multi-label classification using box embeddings. ICLR 2022 Poster (2022)

- 18. Patterson, G., Hays, J.: Coco attributes: Attributes for people, animals, and objects. In: ECCV. pp. 85–100. Springer (2016)
- 19. Pham, K., Kafle, K., Lin, Z., Ding, Z., Cohen, S., Tran, Q., Shrivastava, A.: Learning to predict visual attributes in the wild. In: IEEE CVPR. pp. 13018–13028 (2021)
- Pham, K., Kafle, K., Lin, Z., Ding, Z., Cohen, S., Tran, Q., Shrivastava, A.: Improving closed and open-vocabulary attribute prediction using transformers. In: ECCV (2022)
- Plummer, B.A., Wang, L., Cervantes, C.M., Caicedo, J.C., Hockenmaier, J., Lazebnik, S.: Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In: Proceedings of the IEEE international conference on computer vision. pp. 2641–2649 (2015)
- 22. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: ICML. pp. 8748–8763. PMLR (2021)
- 23. Schuhmann, C., Vencu, R., Beaumont, R., Kaczmarczyk, R., Mullis, C., Katta, A., Coombes, T., Jitsev, J., Komatsuzaki, A.: Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. arXiv preprint arXiv:2111.02114 (2021)
- Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Gradcam: Visual explanations from deep networks via gradient-based localization. In: IEEE ICCV. pp. 618–626 (2017)
- Sharma, P., Ding, N., Goodman, S., Soricut, R.: Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 2556–2565 (2018)
- 26. Yamada, Y., Tang, Y., Yildirim, I.: When are lemons purple? the concept association bias of clip. arXiv preprint arXiv:2212.12043 (2022)
- 27. Yuksekgonul, M., Bianchi, F., Kalluri, P., Jurafsky, D., Zou, J.: When and why vision-language models behave like bag-of-words models, and what to do about it? arXiv preprint arXiv:2210.01936 (2022)
- 28. Zeng, H., Ai, H., Zhuang, Z., Chen, L.: Multi-task learning via co-attentive sharing for pedestrian attribute recognition. In: IEEE ICME. pp. 1–6 (2020)
- Zhang, P., Goyal, Y., Summers-Stay, D., Batra, D., Parikh, D.: Yin and Yang: Balancing and answering binary visual questions. In: IEEE CVPR (2016)
- 30. Zhao, T., Zhang, T., Zhu, M., Shen, H., Lee, K., Lu, X., Yin, J.: Vl-checklist: Evaluating pre-trained vision-language models with objects, attributes and relations (2022). https://doi.org/10.48550/ARXIV.2207.00221, https://arxiv.org/abs/2207.00221
- 31. Zhu, D., Chen, J., Shen, X., Li, X., Elhoseiny, M.: Minigpt-4: Enhancing vision-language understanding with advanced large language models. arXiv preprint arXiv:2304.10592 (2023)