Learning by Correction: Efficient Tuning Task for Zero-Shot Generative Vision-Language Reasoning

Rongjie Li[1*] Yu Wu[1*] Xuming He[1,2‡]
[1]School of Information Science and Technology, ShanghaiTech University
[2]Shanghai Engineering Research Center of Intelligent Vision and Imaging
{lirj2, wuyu1, hexm}@shanghaitech.edu.cn

Abstract

Generative vision-language models (VLMs) have shown impressive performance in zero-shot vision-language tasks like image captioning and visual question answering. However, improving their zero-shot reasoning typically requires second-stage instruction tuning, which relies heavily on human-labeled or large language model-generated annotation, incurring high labeling costs. To tackle this challenge, we introduce Image-Conditioned Caption Correction (ICCC), a novel pre-training task designed to enhance VLMs' zero-shot performance without the need for labeled task-aware data. The ICCC task compels VLMs to rectify mismatches between visual and language concepts, thereby enhancing instruction following and text generation conditioned on visual inputs. Leveraging language structure and a lightweight dependency parser, we construct data samples of ICCC task from image-text datasets with low labeling and computation costs. Experimental results on BLIP-2 and InstructBLIP demonstrate significant improvements in zero-shot image-text generation-based VL tasks through ICCC instruction tuning.

1. Introduction

Vision-language models (VLMs) have demonstrated remarkable performance across a wide range of vision-language (VL) tasks, including image captioning [1, 19, 22, 33, 36], visual recognition [15, 29, 44], image-text retrieval [18, 29], and answering visual questions [1, 19, 22, 33, 36]. Generally, existing VLMs are able to conduct two essential tasks: image-text matching (ITM) and image-text generation (ITG). The contrastive-based ITM aims to model the similarity between vision and text through a shared embedding [15, 18, 29]. In contrast, the generative-based ITG has more flexibility in adapting to various VL tasks. More-

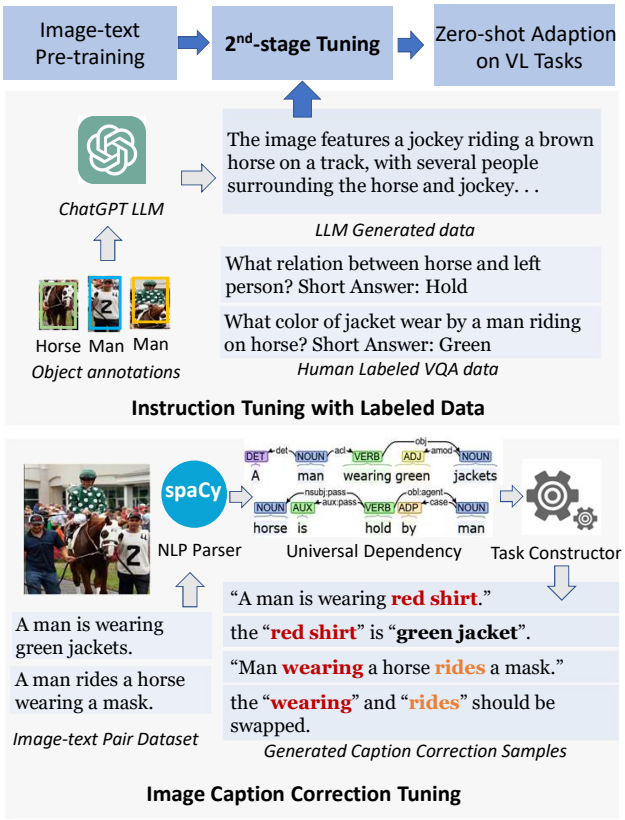

Figure 1. **Illustration of second-stage tuning for zero-shot VL task adaptation comparison.** The instruction tuning in recent works needs human label or LLM-generated data; in contrast, our image caption correction tuning is conducted on unlabeled image-text data with an NLP parser.

*Equal Contribution. This work was supported by NSFC 62350610269, the Shanghai Frontiers Science Center of Human-centered Artificial Intelligence, and the MoE Key Lab of Intelligent Perception and Human-Machine Collaboration (ShanghaiTech University). ‡ denotes corresponding author. Code is available: https://github.com/SHTUPLUS/ICCC_CVPR2024

over, several recent VLM frameworks integrate the LLMs for ITG, which extend the powerful text generation across vision and text modality [1, 8, 19, 22], and allow VLMs to perform zero-shot inference on various VL tasks with impressive performance.

To perform zero-shot inference on VL tasks, the VLMs need to have generalizable text generation capability according to text inputs and concepts from the visual modality. The existing works typically conduct second-stage instruction tuning for pre-trained VLMs with task-oriented data. This improves VLMs for following instructions to generate texts conditioned on visual modality, which ultimately enhances the zero-shot performance on VL-tasks such as InstructBLIP [8] and LLAVA [22], as shown in the upper part of Fig. 1. However, these methods necessitate substantial downstream task data annotation for fine-tuning, which is either human-labeled or generated externally by large language models. This process escalates labor costs throughout the system.

In this study, we introduce a novel pre-training task, Image-Conditioned Caption Correction (ICCC), aimed at enhancing VLMs' performance on zero-shot VL tasks. Our approach leverages the semantic dependency structure of language utilized for second-stage tuning of VLMs, using image-text data without task-specific annotation, as depicted in Fig. 1. By enforcing VLMs to identify and rectify mismatched concepts between visual and language, our method enhances VLMs' capability of generating text from the visual modality. Importantly, the adopted universal semantic dependency [26] ensures comprehensive coverage of various concepts, including objects, their attributes, and interactions between them. Furthermore, we construct the data from unlabeled image-text datasets only with a lightweight dependency parser, which achieves low labeling and computation costs.

Specifically, our pre-training framework first generates the task of image-conditioned text correction in an automatic manner. To this end, we develop a data construction pipeline with two components: the *concept extractor* and the *correction task constructor*. Firstly, the *concept extractor* identifies various concepts from the text modality. With the off-the-shelf dependency parser, it extracts the set of language units by parsing the semantic dependency structure of text. Subsequently, the *correction task constructor* generates samples from the unlabeled image-text data according to language structure and concept set. It swaps or replaces language units according to the extracted concept set, thereby creating concept-mismatched image-text pairs. Thanks to the universality of dependency, this approach allows us to create a wide variety of samples covering diverse visual-language concepts. The resulting text correction task requires VLMs to detect and recover the language units of mismatched concepts (words and phrases) according to the image. Finally, we use the generated samples together with the original image-text data to fine-tune pre-trained VLMs with language modeling objectives.

We conduct extensive experiments on two VLMs, BLIP-2 [19] and InstructBLIP [8], and evaluate the zero-shot performance of ITG on the representative tasks: visual question answering and image caption. Our findings reveal that our proposed method yields substantial improvements in zero-shot generalization based on the initial pre-train model without requiring any manually labeled or LLM-generated data. The main contributions of our work are threefold:

- We introduce a novel image-conditioned text correction fine-tuning strategy for VLMs that enhances their generalization of ITG for VL tasks.
- We developed an automated data construction pipeline that produces large amounts of samples for fine-tuning, all generated from image-text pairs without the need for human annotations or additional LLMs.
- We demonstrated notable improvements in the zero-shot generalization capabilities of VLMs across various VL tasks.

2. Related Work

Generative Vision Language Models With advancements in large-scale pretraining, vision-language models (VLMs) have demonstrated notable zero-shot generalization across various tasks. Unlike contrastive VLMs such as CLIP [29] and ALIGN [15], which focus on image-text similarity scores, generative VLMs output text based on image and text inputs for tasks like visual question answering and image captioning.

In earlier studies, generative VLMs utilized fusion-encoder transformers [4, 20, 32, 41] to simultaneously encode visual and linguistic tokens, and subsequent models [5, 33, 36] integrated visual input information into different architecture language models, giving rise to unified generative VL transformer models. Recently, with advancements in Large Language Models (LLMs) [6, 28, 42, 43], efforts have sought to utilize LLMs' capabilities to project visual input into the language embedding space. BLIP-2 [19] bridges the modality gap using a pre-trained Q-Former. Building upon BLIP-2, MiniGPT-4 [45] and InstructBLIP [8] focus on next-stage instruction tuning to further enhance performance. LLaVA [22] involves GPT-4 [27] in generating instructions and conversations for training. LLaMA-Adapter v2 [11] and LaVIN [24] employ adapters for architecture fine-tuning.

In our study, akin to LLaVA and Instruct-BLIP, we focus on leveraging the benefits of second-stage fine-tuning of pre-trained VLMs. However, unlike these approaches, we aim to achieve this without down-stream tasks data from human annotation and large models generation. Instead, we introduce a novel correction task, construct from the image-

text data with light-weight pipeline.

Language Structure-based Data Augmentation Data augmentation is widely employed in machine learning, with common techniques such as mixup [40], CutMix [39], and RandAugment [7] in computer vision, and back-translation [35], random word editing [34] in natural language processing. In the field of vision and language (VL), an effective approach often involves synthesizing new data using pre-trained models as augmentation [2]. For instance, BLIP [18] leverages the model's initially trained capabilities to generate additional training data, thereby further enhancing the model's performance.

Some researchers in visual-linguistic (VL) studies pursue performance enhancements on challenging tasks by crafting difficult negative samples. For instance, NegCLIP [38] employs a language parser to swap elements between sentences, generating hard negative image-text pairs for contrastive fine-tuning. SVLC [9] achieves a similar outcome by substituting elements in sentences. Our approach leverages linguistic structure to improve image-text generation (ITG) tasks. Unlike image-text matching (ITM) tasks, which focus on learning similarity metrics, generative models encounter difficulties with negative samples. To address this, we introduce a correction task for instruction fine-tuning, using structural language information to construct data samples. This enhances the generalization performance of VLMs in zero-shot ITG-based visual language tasks.

3. Preliminary

Image-text Generation of VLMs Image-text generation, a fundamental task of VLMs, involves generating text from both visual and textual inputs. Specifically, VLMs with parameters $\mathbf{\Theta}_{vlm}$ generate output text from images and textual input in an auto-regressive manner. The visual input is projected into language embedding as visual tokens $\mathbf{Z}$ together with textual input $\mathbf{w} = [w_1, w_2, ..., w_{i-1}]$ and fed into the subsequent LLM to predict the next token w_i of the sequence $\mathbf{w}$.

$$w_i = \mathcal{F}_{vlm}(\mathbf{Z}, w_1, w_2, ..., w_{i-1}; \mathbf{\Theta}_{vlm}). \quad (1)$$

Second-stage Tuning for Zero-shot Generation While generative image-text pre-training provides VLMs with aligned vision-language representations, effective instruction-following of image-text generation for diverse zero-shot VL tasks is crucial. Recent works address this issue by leveraging either human-labeled [8] or large language model-generated task-oriented data [22] for instruction tuning. Concretely, in the second stage of tuning, the VLMs are optimized with the same objective of generative image-text pre-training with task-oriented data:

$$\arg\max_{\mathbf{\Theta}_{vlm}} \sum_{i=1}^{K} \log P(w_i|\mathbf{Z}, w_1, w_2, ..., w_{i-1}; \mathbf{\Theta}_{vlm}). \quad (2)$$

4. Our Approach

In this section, we introduce our proposed Image Conditional Caption Correction (ICCC) task for second-stage tuning, which constructs data from unlabeled data with low resource consumption. We first provide a *Task Definition* of ICCC in Sec. 4.1. Then we introduce the data construction pipeline for ICCC, which includes two modules: the *Concept Extractor* and the *Correction Data constructor*. The *concept extractor* parses the text structure and extracts the concept set for the following data construction, discussed in Sec. 4.2. Subsequently, the *correction data constructor* generates data by augmenting text structure with the extracted concept set, described in Sec.4.3, Finally, Sec. 4.4 outlines our training and inference procedures for incorporating our task into VLM pre-training.

4.1. Task definition

As illustrated in Fig. 2, the ICCC task involves identifying and correcting language units of mismatched concepts of caption. To define the ICCC task, we need to define the concept set, how we perturb to produce concept-mismatched samples, and how the perturb operation changes the language structure.

For each sentence, concepts are composed of linguistic units representing their semantic meaning. These linguistic units are categorized into five types based on semantics and language granularity: $\mathcal{E} = \{$*entity phrase, predicate phrase, attribute phrase, noun word, verb word*$\}$. The *entity phrase*, *noun word*, *attribute phrase* represent the object-level semantic, and *predicate phrase*, *verb word* represent the composition-level semantic. The *phrase* and *word* represent different levels of language granularity.

There are two operations for perturbing the language structure: $\{\mathsf{replace}, \mathsf{swap}\}$. The replace involves substituting a concept with another one of the same type from the image-text pair dataset, while the swap involves swapping the positions of two concepts of the same type within the original caption. More details are introduced in Sec. 4.3. Consequently, replace focuses on modeling the intrinsic meaning of individual concepts, while swap prioritizes the order of compositional concepts within sentences. Overall, the task is to use linguistic unit modifications of image-text pairs, providing universal concepts aligned between vision and language, which improves text generation capabilities across two modalities.

4.2. Concept Extractor

As mentioned earlier, the data of ICCC is generated by perturbing concepts in $\mathcal{E}$ according to the structure of sen-

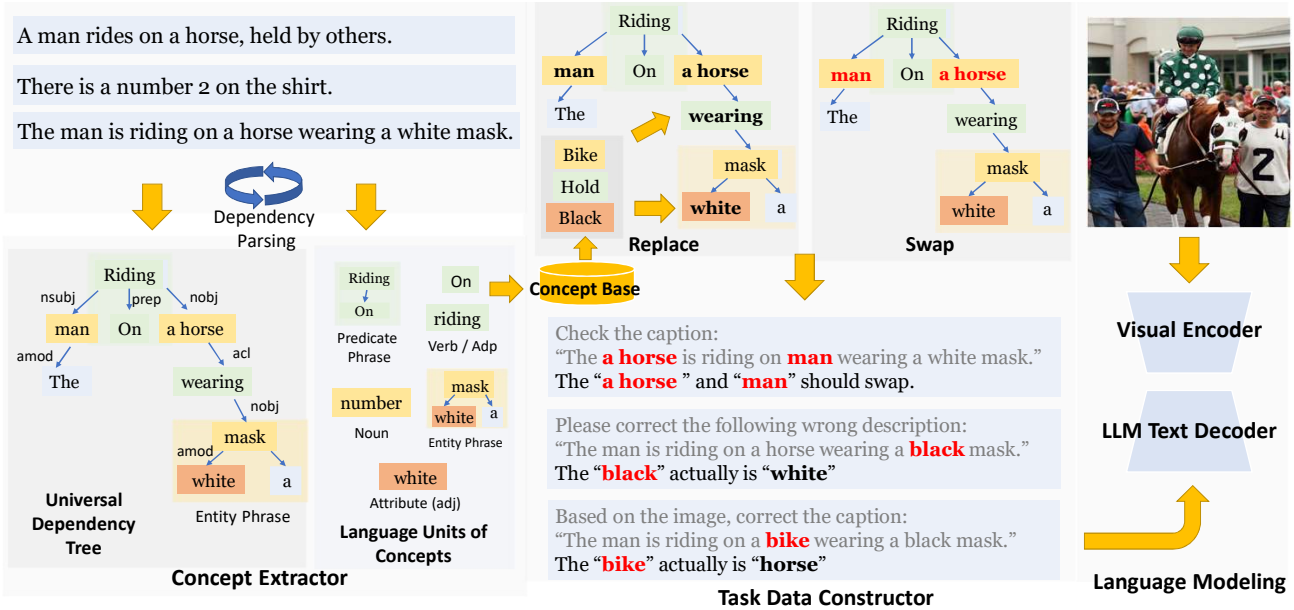

Figure 2. **Illustration of the overall pipeline of ICCC.** The concept extractor parses the sentence to obtain linguistic units of concepts. The task data constructor aims to produce the sample according to the sentence structure with the "replace" and "swap" operations. Finally, the generated ICCC data is used for image-to-text generative training for VLMs.

tences. The concept extractor takes captions as input and parses the dependency to identify linguistic units of concepts, which the following task constructor then uses to generate data samples. The concept extractor module has three processes: (1) universal dependency parsing; (2) linguistic unit selection and grouping; (2) concept collection.

Universal Dependency Parsing We use the off-the-shelf universal dependency parser implemented by the spaCy [13] software library to extract the dependency structure of the original caption. As shown in Fig. 2, the dependency parser translates the sentence into a universal dependency tree [26]. The node of the dependency tree is the minimal linguistic unit $u_{[pos]}$, which has Part-of-Speech (POS) tags to indicate its grammatical role. The edge represents the dependency relation between linguistic units $r_{[REL]}$, which also has the relation type (REL). The dependency structure provides us with an indication of how linguistic units organize to represent the concept of a sentence.

Linguistic Unit Selection and Grouping We extract the linguistic unit set of concept $U_{concept}$ within $\mathcal{E}$ by grouping and selecting $u_{[pos]}$. To achieve this, we design a heuristic method by traversing the dependency tree to select the corresponding $u_{[pos]}$. Firstly, we extract the concept represented by the concept of a word-level linguistic unit according to the POS tag:

1. $U_{noun\ word} : u_{[noun]}$
2. $U_{verb\ word} : u_{[verb]}$

Based on such word-level concepts $u_{[pos]}$, we further extract linguistic units of phrase-level concepts, such as *entity phrase, predicate phrase, attribute phrase*, by grouping the $u_{[pos]}$ into $U_{concept}$ according to the $r_{[REL]}$ associated with word-level concept units:

1. $U_{entity\ phrase}$: $u_{[adj]}$ and $U_{[det]}$ adjacent to $U_{[noun]}$
2. $U_{predicate\ phrase}$: all $u_{[pos]}$ within two $u_{[noun]}$
3. $U_{attribute\ phrase}$: all $u_{[amod]}$ adjacent to $u_{[noun]}$

To this end, we extract the linguistic units of each concept type $U_{concept}$ from the text according to the dependency structure.

Concept Collection For each sentence from the image-text dataset, we collect all $U_{concept}$ according to their concept type, respectively. We merge the $U_{noun\ word}$ with the $U_{entity\ phrase}$ and the $U_{verb\ word}$ with the $U_{predicate\ phrase}$ into the same concept type, respectively. This global-level concept base stores all the concepts that occurred in the dataset. We filter them by frequency to remove infrequent concepts, which could be extracted from low-quality captions or parsing errors, and the most frequent, which could be a trivial concept or language bias that occurred uniformly.

4.3. Correction Task Data Constructor

The correction task data constructor uses the extracted concept structure to generate data samples with multi-level concept mismatch captions. The data constructor takes the input text, which has identified concepts of language units and concept-based units, and produces the augmented text with expected corrections.

Specifically, the data constructor perturbs the initial lin-

guistic structure by predefined operations: replace, swap, as shown in Fig. 2. Specifically, the replace randomly selects the concept of the original text and replaces it with an external $U_{concept}$ that has the same semantic type but does not occur in the current text from the concept base. For instance, the *entity phrase* can be replaced by all object-level concepts, such as *noun word* or *entity phrase*. The swap operation, instead of replacing concept by concept, swaps the two linguistic unit sets with the same concept type in the text. This perturbation operation provides the order mismatch of the concept. To calibrate those two operations, we randomly select one by Bernoulli distribution with a preset parameter $p_s \in [0, 1]$ to decide which type of operation is used for perturbation. In those cases where a sentence doesn't have two of the same concept-type linguistic units, we simply use the replace operation.

After the perturb operation, we map back the tree structure sentence into sequence and construct the caption correction samples, which are composed of the following components: correction instruction prompt, perturbed caption, and correcting description, as shown in Fig. 2. The correction instruction prompt is selected from a template base, and the correcting description includes the mismatch concept and the correct ones.

4.4. Training and Inference

The data generated by ICCC is utilized in the second-stage tuning process to improve VLMs, as introduced in Sec 3. We combine ICCC data samples with original image-text pairs for training to prevent focusing too much on the specific task and catastrophic forgetting. The proportion of ICCC data samples to original pairs in each training batch is determined by a hyper-parameter p_c. The overall pre-training objective remains unchanged from the initial image-text generation pre-training. This design enables VLMs to grasp the alignment of concepts and facilitate instruction following for various downstream generative VL tasks, enhancing their performance across different tasks. During inference, VLMs conduct standard image-to-text generation for VL tasks, consistent with previous generative VLMs [8, 19].

5. Experiments

In this section, we thoroughly explore the effectiveness of our proposed training task. We detail the experimented models and implementation in Sec. 5.1, conduct primary zero-shot evaluations in Sec. 5.2, present ablation studies in Sec. 5.3, and offer qualitative insights in Sec. 5.4.

5.1. Experiments Configuration

Models and Training Data In the realm of generative VLMs, we select BLIP-2 [19] and InstructBLIP [8], representing different pre-training data paradigms. BLIP-2 utilizes image-text pair data without instructions, employing a frozen image encoder and a large language model. It integrates a lightweight querying transformer for mapping visual information into the language embedding space. Instead, InstructBLIP incorporates instruction tuning, enhancing the architecture of pre-trained BLIP-2 with an instruction-aware visual feature extractor. It undergoes complementary training with additional task-specific instruction-tuning data.

In our model setup, we explore three variants of BLIP-2 to assess the generality of our training method across different LLM architectures. They share the same image encoder (ViT-G/14 from EVA-CLIP [10]) but employ distinct frozen LLMs: OPT [42] with 2.7B and 6.7B parameters, and FlanT5-XL [15] with 3B parameters. For InstructBLIP, we experiment with the Vicuna-7B [43] version. Initializing both models with pre-trained parameters, we freeze the vision encoder and LLM, focusing solely on training their Q-former and fully connected projection network.

We construct ICCC samples using our proposed method on the COCO Caption [3] and Visual Genome (VG) Caption [16] datasets, totaling approximately 1 million image-text pairs. These samples are then utilized for language modeling training and image captioning in a second-stage training paradigm outlined in Sec. 4.4.

Implementation Details We implement and evaluate our method using the LAVIS library [17], and mainly followed the training setup for the original models. The AdamW [23] optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and a weight decay of 0.05 was employed. We used a linear warmup of the learning rate over the initial 1,000 steps, increasing from 10^{-8} to 10^{-5}, followed by a cosine decay with a minimum learning rate of 0. Batch sizes varied across models: 64 for BLIP-2 OPT2.7B and BLIP-2 FlanT5-XL, 28 for BLIP-2 OPT6.7B, and 24 for InstructBLIP. The images are resized to size 224×224, and we apply random resized cropping and horizontal flipping augmentations. All training spanned a maximum of 20,000 iterations, with model performance validated every 1,000 iterations. Each training process utilized four Nvidia A40 (40G) GPUs, completed within a day.

Regarding the hyperparameters, we conducted comparison experiments. We set (p_c, p_s) as $(0.3, 0.15)$, $(0.3, 0)$, $(0.01, 0.2)$, and $(0.3, 0.3)$. for BLIP-2 OPT2.7B, BLIP-2 OPT6.7B, BLIP-2 FlanT5-XL, and InstructBLIP, respectively. Further details will be demonstrated in Sec. 5.3.

5.2. Zero-shot Evaluations

We present zero-shot evaluation results in Tab. 1 and Tab. 2 for BLIP-2 and InstructBLIP experiments. Our second-stage training strategy ICCC consistently improves zero-shot performance across various tasks and datasets, like visual question answering (VQA) and image captioning (IC),

BLIP-2	GQA	OK-VQA	VQAv2	VSR	NoCaps		
					B@4	S	C
OPT2.7B	33.5	26.6	51.9	**48.3**	43.6	13.8	105.7
OPT2.7B w/ ICCC	**38.9**	**29.5**	**54.3**	47.8	**46.0**	**14.3**	**111.9**
OPT6.7B	35.5	30.7	52.6	48.5	41.5	13.0	101.4
OPT6.7B w/ ICCC	**38.2**	**31.7**	**58.8**	**51.5**	**44.1**	**13.6**	**106.9**
FlanT5-XL	44.0	40.7	63.1	63.4	42.2	13.3	103.1
FlanT5-XL w/ ICCC	**44.6**	**41.0**	**64.0**	**64.2**	**43.9**	**13.6**	**106.0**

Table 1. **The zero-shot evaluation results on BLIP-2 experiments.** For three VQA datasets, we report top-1 accuracy (%) on the testdev set of GQA [14], VSR [21], the test set of OK-VQA [25], and the validation set of VQAv2 [12]. For IC, we report metrics of BLUE@4 (B@4), CIDEr (C), and SPICE (S) on the validation set of NoCaps.

Model	GQA	VSR	NoCaps
InstructBLIP	48.4	61.1	14.2
InstructBLIP w/ ICCC	**49.8**	**63.1**	**15.7**

Table 2. **The zero-shot evaluation results on InstructBLIP experiments.** We report the SPICE score for NoCaps.

demonstrating enhanced zero-shot generalization.

The experimental results on BLIP-2 (Tab. 1) demonstrate the effectiveness of ICCC on models pre-trained by image-text pairs. Notable findings include:

- ICCC yields consistent improvements for BLIP-2 OPT2.7B, BLIP-2 OPT6.7B, and BLIP-2 FlanT5-XL. Specifically, the BLIP-2 OPT2.7B with ICCC shows a significant enhancement in GQA (+5.4%), while the BLIP-2 OPT6.7B improves considerably in VQAv2 (+6.2%). These results emphasize ICCC's ability to refine vision-conditioned language generation, irrespective of LLM sizes and architectures.
- In captioning tasks, SPICE metrics highlight improved accuracy at the structured scene level, including VL relations and attributes. Our task design, focusing on learning ITG generalizations for various concept roles, not only enhances object recognition accuracy but also improves understanding of object attributes and relations.
- In the InstructBLIP experiment results shown in Tab. 2, we note a consistent improvement in zero-shot evaluation benchmarks, even when the original model heavily relies on instruction tuning. This indicates the effectiveness of our approach across diverse pre-training image-text data, offering a valuable complementary method for augmenting visual-linguistic knowledge.
- We emphasize the Visual Spatial Reasoning (VSR) benchmark [21], created to evaluate a model's grasp of spatial relationships for vision-language reasoning. These enhancements stem from the varied concept data samples in our ICCC pre-training, which are able to encompass a broad spectrum of vision-language relational concepts.
- Additionally, we assess the effectiveness of ICCC on another VLM (LLAVA [22]) and various vision-language reasoning datasets, including ScienceQA-IMG [31], MM-VET [37], and hallucinations [30]. Further experimental results can be found in the supplementary material.

	Data Source			GQA	OK-VQA	VQAv2	NoCaps		
	C+V	VQAv2	w/ ICCC				B@4	S	C
1	-	-	-	33.5	26.6	51.9	43.6	13.8	105.7
2	✓	-	-	33.0	26.3	49.7	45.6	13.9	108.1
3	✓	-	✓	**38.9**	**29.5**	**54.3**	**46.0**	**14.3**	**111.9**
4	✓	✓	-	41.3	38.0	-	**45.2**	12.6	106.4
5	✓	✓	✓	**42.8**	**39.1**	-	45.0	**14.0**	**108.2**

Table 3. **Ablation study on our method upon different second-stage training data types.** C+V represents COCO and VG caption data. Since our focus is on zero-shot performance, we omit the results on the VQAv2 validation set after training with the VQAv2 training set.

5.3. Ablation Study

Effectiveness of Correction Task In Tab. 3, we compare our approach with alternative second-stage training methods, focusing on the BLIP-2 OPT2.7B model. We explore two types of second-stage training: one utilizing only COCO and VG caption datasets (lines 2 and 3), and the other incorporating the VQAv2 training set (lines 4 and 5) to assess the impact of instruction tuning. Our approach exclusively targets caption datasets, maintaining consistent task sample construction parameters.

Experimental results indicate that our training approach consistently outperforms conventional second-stage methods, regardless of the inclusion of VQA data for instruction-tuning. This underscores the efficacy of our task, emphasizing its superiority over traditional ITG second-stage training strategies.

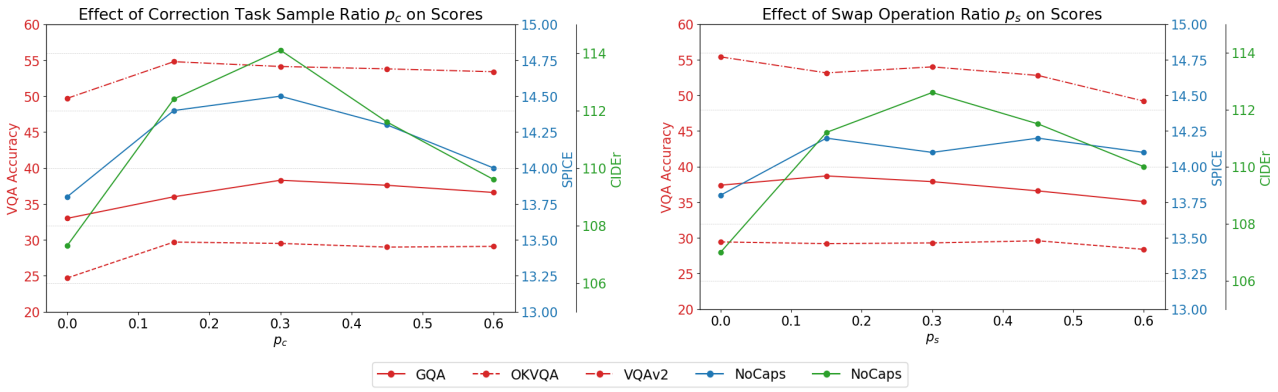

Figure 3. **Hyper-parameter searching on p_c and p_s.**

Ablation for Task Construction In Tab. 4, we demonstrate the necessity of including different types of concepts to edit for constructing mismatched captions. The experiments were conducted on the BLIP-2 OPT2.7B model, maintaining consistent hyper-parameters. Results indicate that using concepts of different natures for task construction yields complementary effects.

- In the first part, we investigate influence of language granularity, focusing on tasks constructed around words and phrases. As shown in lines 3–4 of Tab.4, our findings reveal the significance of phrase-level concepts for the VQA task, while word-level concepts aid in refining the model's understanding of individual words, thereby improving captioning accuracy metrics such as BLUE@4 and CIDEr.
- In the subsequent section (lines 5–6), we delve into the roles of various concepts in semantics, exploring the impact of tasks centered on entity-level, relation-level, and attribute-level concepts. Referencing Tab.4, our analysis reveals that focusing solely on learning at the entity level yields strong performance in VQA tasks but may sacrifice accuracy in comprehending relations and attributes, thereby resulting in sub-optimal image captioning performance.
- We conducted an ablation study to assess the impact of instructions and language structure (line 7, Tab.4). We retained the instructions of the ICCC task while randomly altering words within the sentences. Surprisingly, the addition of correction instructions and random language masking did not enhance the generalization capability of VLMs on downstream tasks. This highlights the importance of language structure in instruction tuning.

In conclusion, the diversity in concept extraction allows our method to perform well on various zero-shot generation tasks, demonstrating strong generality.

Hyper-parameter Selection The hyperparameters p_c and p_s are pivotal in adjusting the distribution of task samples

	Type	GQA	OK-VQA	VQAv2	NoCaps		
					B@4	S	C
1	none	33.0	26.3	49.7	45.6	13.9	108.1
2	all	**38.9**	29.5	54.3	46.0	14.3	111.9
3	*noun*, *verb*	36.4	**30.7**	52.1	46.0	14.3	**114.3**
4	*ent*, *pred*, *attr*	38.3	27.5	**54.9**	45.2	**14.7**	111.3
5	*noun*, *ent*	38.5	29.8	54.8	46.5	14.4	112.4
6	*verb*, *pred*, *attr*	36.0	29.6	54.1	**46.9**	14.5	114.1
7	random	33.0	26.2	48.6	44.1	14.5	107.1

Table 4. **Ablation study on editing different subsets of concept types for mismatched caption construction.**

in our approach. We conducted a comprehensive hyperparameter search for each experimental model, analyzing the impact of these parameters on model performance. Fig. 3 illustrate the performance variations of BLIP-2 OPT2.7B after second-stage training under different hyperparameter settings:

- In the left of Fig. 3, with p_s fixed at 0.1, we study the influence of different correction task sample proportions by varying p_c at intervals of 0.15.
- In the right of Fig. 3, with p_c fixed at 0.3, we explore the impact of different swap operation proportions on mismatched caption sampling by varying p_s at intervals of 0.15.

In our experiments, we present several key findings:

- From both figures, it is evident that the impact of hyperparameters p_c and p_s on model performance follows a consistent trend, initially increasing before decreasing. This phenomenon arises due to the risk of language bias when incorporating excessive correction task samples, particularly those generated through swap operations.
- Notably, the saturation point of p_s is reached early, suggesting that the model can easily identify and correct mismatched captions created through swap operations. Despite this, proper inclusion of such samples proves beneficial for overall performance enhancement, particularly

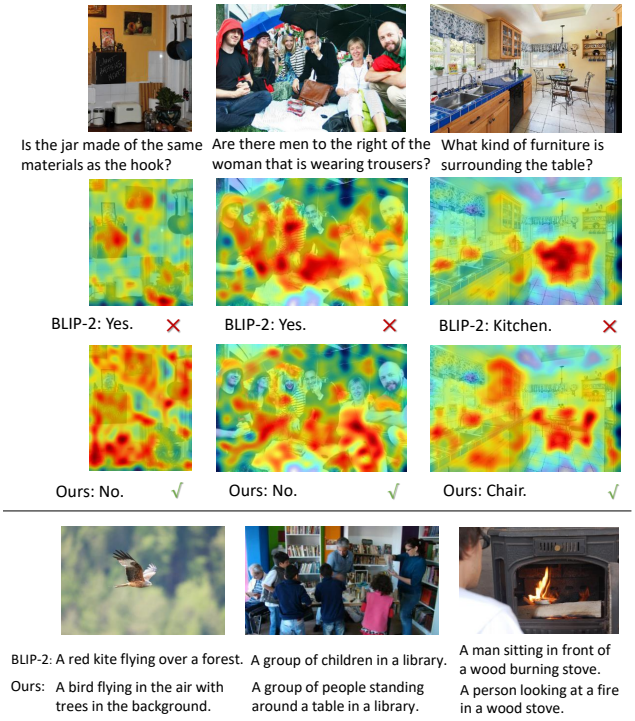

Figure 4. **Visualization results include model output examples and attention gradients on images.** The first block illustrates three examples from the GQA testdev set, while the second block showcases three examples from the NoCaps validation set. With our training, the model demonstrates improved accuracy in focusing on prompt-relevant image regions. Additionally, it generates captions with more detailed descriptions of scenes and actions.

in image captioning tasks.

- Furthermore, our investigation reveals that BLIP-2 FlanT5-XL tends to overfit on our task samples, possibly attributed to its encoder-decoder architecture. This results in the Q-former learning an overfitted encoding of soft prompts associated with task text patterns. Nonetheless, even minimal incorporation of correction task samples ($p_c = 0.01$) effectively serves our training objectives while mitigating the risk of overfitting.

5.4. Qualitative Results

To provide a more intuitive understanding of the effects of our training, we show some qualitative model output examples in Fig. 4. We demonstrate the differences between the model outputs and intermediate results on the VQA and IC tasks before and after our second-stage training respectively. The heat maps show the gradient of the output logit corresponding to the ground truth token with regard to the image attention, which reflects the potential visual contribution of the model output.

According to the outputs for VQA, the results reveal that the proposed correction task training produces interpretable improvements to VQA responses. Our model demonstrates increased accuracy in locating objects referenced in prompts and exhibits a more precise understanding of compositional concepts, avoiding biases toward prominent objects. For instance, in the first example, our model focuses on the "hook", which the original model misses. In the second example, our model correctly considers the composed concept of "the woman that is wearing trousers". Lastly, our model interprets the relational concept "surrounding the table" and directs attention appropriately toward the relevant region, rather than the table itself. Overall, our findings indicate that this training methodology enhances VLMs' capacity for interpreting corresponding visual concepts expressed through language.

Examples from our experiments with IC illustrate this improved ability to correct misidentified objects and provide richer, more fine-grained descriptions of object relationships and actions. For instance, our model correctly identifies the wrong "red kite" concept in the first case and provides more nuanced details about the scene, such as "standing around a table" and "looking". We posit that the primary source of improved overall captioning performance lies in the generation of more accurate and comprehensive concepts in captions.

Conclusively, our study indicates that the proposed training method potentially enhances generative VLMs' capacity for aligning concepts across multiple types and granularities in visual and language modalities. We believe this insight could shed light on the future development of cost-effective, multi-granularity, and structured generative VL pre-trained models.

6. Conclusion

In this work, we propose the image-conditioned text correction task for enhancing zero-shot text generation with unlabeled data. In this task, VLMs requires to identify and correct the error in accordance with the vision modality via text generation. We propose a scalable and cost-effective data construction framework for generating the image-text pair for this task by utilizing the inherent structure of language. The experimental results indicate that the implementation of our training framework substantially improves the ability of VLMs to generalize across a range of VL tasks involving image-to-text generation.

Discussion of Limitation Due to the limitation of the computing resource, extending our approach to larger datasets and models remains unexplored. Future work should explore its application to more extensive datasets and diverse large vision-language models.

References

[1] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in Neural Information Processing Systems*, 35:23716–23736, 2022. 1, 2

[2] Paola Cascante-Bonilla, Khaled Shehada, James Seale Smith, Sivan Doveh, Donghyun Kim, Rameswar Panda, Gul Varol, Aude Oliva, Vicente Ordonez, Rogerio Feris, et al. Going beyond nouns with vision & language models using synthetic data. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20155–20165, 2023. 3

[3] Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*, 2015. 5

[4] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. Uniter: Universal image-text representation learning. In *European conference on computer vision*, pages 104–120. Springer, 2020. 2

[5] Jaemin Cho, Jie Lei, Hao Tan, and Mohit Bansal. Unifying vision-and-language tasks via text generation. In *International Conference on Machine Learning*, pages 1931–1942. PMLR, 2021. 2

[6] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*, 2022. 2

[7] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 702–703, 2020. 3

[8] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning, 2023. 2, 3, 5

[9] Sivan Doveh, Assaf Arbelle, Sivan Harary, Eli Schwartz, Roei Herzig, Raja Giryes, Rogerio Feris, Rameswar Panda, Shimon Ullman, and Leonid Karlinsky. Teaching structured vision & language concepts to vision & language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2657–2668, 2023. 3

[10] Yuxin Fang, Wen Wang, Binhui Xie, Quan Sun, Ledell Wu, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. Eva: Exploring the limits of masked visual representation learning at scale. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19358–19369, 2023. 5

[11] Peng Gao, Jiaming Han, Renrui Zhang, Ziyi Lin, Shijie Geng, Aojun Zhou, Wei Zhang, Pan Lu, Conghui He, Xiangyu Yue, et al. Llama-adapter v2: Parameter-efficient visual instruction model. *arXiv preprint arXiv:2304.15010*, 2023. 2

[12] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6904–6913, 2017. 6

[13] Matthew Honnibal and Ines Montani. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear, 2017. 4

[14] Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6700–6709, 2019. 6

[15] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*, pages 4904–4916. PMLR, 2021. 1, 2, 5

[16] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123(1):32–73, 2017. 5

[17] Dongxu Li, Junnan Li, Hung Le, Guangsen Wang, Silvio Savarese, and Steven C.H. Hoi. LAVIS: A one-stop library for language-vision intelligence. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 31–41, Toronto, Canada, 2023. Association for Computational Linguistics. 5

[18] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning*, pages 12888–12900. PMLR, 2022. 1, 3

[19] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*, 2023. 1, 2, 5

[20] Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, et al. Oscar: Object-semantics aligned pre-training for vision-language tasks. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXX 16*, pages 121–137. Springer, 2020. 2

[21] Fangyu Liu, Guy Emerson, and Nigel Collier. Visual spatial reasoning. *Transactions of the Association for Computational Linguistics*, 11:635–651, 2023. 6

[22] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*, 2023. 1, 2, 3, 6

[23] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2018. 5

[24] Gen Luo, Yiyi Zhou, Tianhe Ren, Shengxin Chen, Xiaoshuai Sun, and Rongrong Ji. Cheap and quick: Efficient vision-language instruction tuning for large language models. *arXiv preprint arXiv:2305.15023*, 2023. 2

[25] Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. Ok-vqa: A visual question answering benchmark requiring external knowledge. In *Proceedings of the IEEE/cvf conference on computer vision and pattern recognition*, pages 3195–3204, 2019. 6

[26] Joakim Nivre, Marie-Catherine De Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajic, Christopher D Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, et al. Universal dependencies v1: A multilingual treebank collection. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1659–1666, 2016. 2, 4

[27] R OpenAI. Gpt-4 technical report. *arXiv*, pages 2303–08774, 2023. 2

[28] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35: 27730–27744, 2022. 2

[29] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 1, 2

[30] Anna Rohrbach, Lisa Anne Hendricks, Kaylee Burns, Trevor Darrell, and Kate Saenko. Object hallucination in image captioning. *arXiv preprint arXiv:1809.02156*, 2018. 6

[31] Tanik Saikh, Tirthankar Ghosal, Amish Mittal, Asif Ekbal, and Pushpak Bhattacharyya. Scienceqa: A novel resource for question answering on scholarly articles. *International Journal on Digital Libraries*, 23(3):289–301, 2022. 6

[32] Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. *arXiv preprint arXiv:1908.07490*, 2019. 2

[33] Peng Wang, An Yang, Rui Men, Junyang Lin, Shuai Bai, Zhikang Li, Jianxin Ma, Chang Zhou, Jingren Zhou, and Hongxia Yang. Ofa: Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. In *International Conference on Machine Learning*, pages 23318–23340. PMLR, 2022. 1, 2

[34] Jason Wei and Kai Zou. Eda: Easy data augmentation techniques for boosting performance on text classification tasks. *arXiv preprint arXiv:1901.11196*, 2019. 3

[35] Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. Unsupervised data augmentation for consistency training. *Advances in neural information processing systems*, 33:6256–6268, 2020. 3

[36] Zhengyuan Yang, Zhe Gan, Jianfeng Wang, Xiaowei Hu, Faisal Ahmed, Zicheng Liu, Yumao Lu, and Lijuan Wang. Unitab: Unifying text and box outputs for grounded vision-language modeling. In *European Conference on Computer Vision*, pages 521–539. Springer, 2022. 1, 2

[37] Weihao Yu, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Zicheng Liu, Xinchao Wang, and Lijuan Wang. Mm-vet: Evaluating large multimodal models for integrated capabilities. *arXiv preprint arXiv:2308.02490*, 2023. 6

[38] Mert Yuksekgonul, Federico Bianchi, Pratyusha Kalluri, Dan Jurafsky, and James Zou. When and why vision-language models behave like bags-of-words, and what to do about it? *arXiv e-prints*, pages arXiv–2210, 2022. 3

[39] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6023–6032, 2019. 3

[40] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017. 3

[41] Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. Vinvl: Making visual representations matter in vision-language models. *arXiv preprint arXiv:2101.00529*, 1(6): 8, 2021. 2

[42] Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*, 2022. 2, 5

[43] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric. P Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena, 2023. 2, 5

[44] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16816–16825, 2022. 1

[45] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023. 2