Segment and Caption Anything

Xiaoke Huang[1†] Jianfeng Wang[2] Yansong Tang[1*] Zheng Zhang[2]
Han Hu[2] Jiwen Lu[3] Lijuan Wang[2] Zicheng Liu[4]

[1]Shenzhen International Graduate School, Tsinghua University [2]Microsoft
[3]Department of Automation, Tsinghua University [4]Advanced Micro Devices

{hxk21@mails.,tang.yansong@sz.,lujiwen@}tsinghua.edu.cn
{jianfw,zhez,lijuanw}@microsoft.com ancientmooner@gmail.com zicheliu@amd.com

Abstract

We propose a method to efficiently equip the Segment Anything Model (SAM) with the ability to generate regional captions. SAM presents strong generalizability to segment anything while is short for semantic understanding. By introducing a lightweight query-based feature mixer, we align the region-specific features with the embedding space of language models for later caption generation. As the number of trainable parameters is small (typically in the order of tens of millions), it costs less computation, less memory usage, and less communication bandwidth, resulting in both fast and scalable training. To address the scarcity problem of regional caption data, we propose to first pre-train our model on objection detection and segmentation tasks. We call this step weak supervision pretraining since the pre-training data only contains category names instead of full-sentence descriptions. The weak supervision pretraining allows us to leverage many publicly available object detection and segmentation datasets. We conduct extensive experiments to demonstrate the superiority of our method and validate each design choice. This work serves as a stepping stone towards scaling up regional captioning data and sheds light on exploring efficient ways to augment SAM with regional semantics. The project page, along with the associated code, can be accessed via the following link.

1. Introduction

Teaching machines to understand the visual world with natural languages has been a long-standing problem in computer vision [30, 74, 76]. Image captioning is one of the topics that require the machine to perceive and describe images in human languages [34, 37]. With the wave of deep learning [24, 39], enormous efforts [42, 43, 84, 96] have been devoted to pushing its frontier in terms of model architectures, training data, training techniques, *etc*. However, much less work has been devoted to the regional captioning [33, 56, 89, 104], in which models describe the regions instead of the entire image.

Building an intelligent system that follows human intent is an emerging research topic, as evidenced by the rapid progress of large foundation models [7, 31, 65, 80, 99]. Major breakthroughs have been made in language modeling [7, 62, 78, 81], where the foundation language models are fine-tuned to follow the instructions of users with both instruction supervision [62, 78] and human feed-

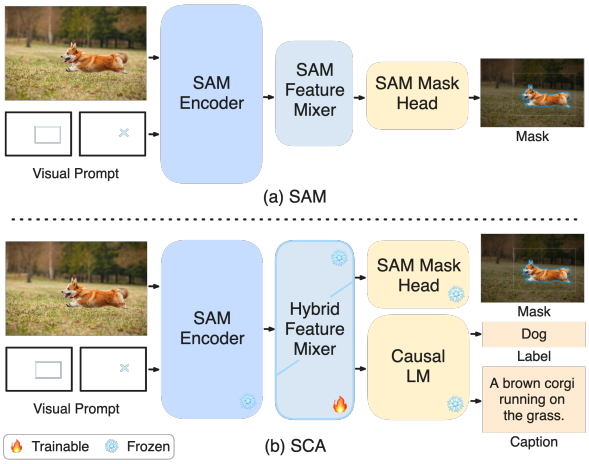

Figure 1. SCA (*b*) is a lightweight augmentation of SAM (*a*) with the ability to generate regional captions. On top of SAM architecture, we add a pre-trained language model which is frozen, and a lightweight hybrid feature mixture. Despite the small number of trainable parameters, the region-specific features are learned to align with the embedding space of the language model for regional caption generation.

[†]Work was done when the author interned at Microsoft.
[*]Corresponding.

back [60, 81]. The idea is further developed in multi-modal language model [53, 109], text-to-image generation [69, 71, 97], and interactive segmentation [35]. *Segment Anything Model* (SAM) [35] is an interactive segmentation system, that successfully scales the mask data to a billion. Such data scale enables stronger generalizability in segmentation given the visual prompts. However, the data contain *no semantic labels* thus the model is incapable of semantic understanding.

We propose a method to efficiently equip SAM with the ability to generate regional captions. We marry SAM with causal language models [7, 64, 80] by introducing a lightweight *hybrid feature mixture* which stacks a text feature mixture on top of the SAM feature mixture. The hybrid feature mixture extracts regional features for downstream caption predictions via self- and cross-attention [82]. We *solely* optimize the text feature mixer and leave the other network modules (*i.e.* SAM's encoder, SAM feature mixer, the language model) untouched. During training, the region-specific features are aligned with the embedding space of language models for later caption generation. As the number of trainable parameters is small (typically in the order of tens of millions), it costs less computation, less memory usage, and less communication bandwidth, resulting in both *fast and scaleable* training. Fig. 1 provides a system overview.

However, there is limited data available for training regional captioning models [36, 98]. For example, One commonly used dataset, Visual Genome (VG) [36] contains up to 100K images. In contrast, SAM [35] used a dataset that contains more than 11M images and 1B masks. Inspired by the effective deployment of weak supervision [44, 103, 111], we introduce a weak supervision pretraining step to leverage the publicly available object detection and segmentation datasets. Specifically, we pre-train the text feature mixer on Objects365 [72] detection data and COCO-Panoptic [50] segmentation data, which consist of 1.8M images. Finally, the text feature mixer is finetuned on the VG regional captioning data.

We have conducted extensive experiments to demonstrate the effectiveness of our method and validate each design choice. Our method achieves state-of-the-art performance on the VG [36] benchmark with 149.8 CIDEr-D, 17.5 METEOR, and 31.4 SPICE. We believe this work serves as a stepping stone towards scaling up regional captioning data [6, 35, 57] and sheds light on exploring efficient approaches to augment a segmentation model like SAM with regional semantics.

2. Related Works

Object detections, segmentations, and interactive segmentations. The field of object detection has evolved from CNN-based methods [19, 23, 55, 55, 67, 68, 79] to transformer-based models [8, 41, 54, 82, 102, 110]. The transformer architecture has shown versatility across modalities, facilitating tasks like open-world detection [31, 65, 99, 100]. Similar architectural trends are observed in segmentation tasks [11, 12, 26]. Recent works have also integrated vision-language pre-training for open-world segmentation [17, 22, 40, 48, 90, 94]. Interactive segmentation [25, 66, 70] is a sub-task with unique challenges that can be tackled by transformer-based models like SAM [35]. This paper extends SAM to region-level understanding using additional tokens and transformer layers.

Image captioning and dense captioning. Image captioning involves generating textual descriptions for images by combining vision and language models [16, 18, 27, 31, 65, 77, 99]. Early methods employed CNN and LSTM [34], while recent works leverage transformers [1, 13, 15, 43, 84] and large language models [7, 62, 80, 81]. These models can follow user instructions, demonstrating abilities like visual reasoning and question answering. Dense captioning [33, 36, 45, 56, 73, 89, 91, 95] extends image captioning to region-level, combining detection with generation. Despite its simultaneous development with image captioning, its evaluation metrics improvement has been slow due to the compounded difficulty of localization and generation. This work assumes localization proposals as given inputs and focuses on region captioning.

Scaling region understanding systems. Tremendous progress has been made in natural language processing and vision domains by training large models on massive datasets, with scaling laws illustrating the relationship between computational budgets, data size, and performance [1, 3, 13, 29, 31, 58, 65, 99]. This trend is also observed in region-level understanding systems, where weak-supervision methods like self-training and proxy training losses are used to scale up data [4, 44, 92, 93, 101, 105, 108]. [57] and [35] show the importance of scaling in vision tasks by reaching the scale of billions of samples. However, region-level understanding is still underexplored due to the limited data scale. The current dataset, Visual Genome [36], is small, leading to poor alignment and generalizability. This work aims to explore the scaling property in generative region-level understanding using weak supervision from detection [38, 50, 72, 85] and leaves image captioning supervision [10, 59, 75] and self-training [42, 57] for future exploration.

Concurrent works. Recent progress in Large Language Model (LLM) and interactive segmentation has spurred several concurrent works in region-level understanding. Without training, Caption Anything [86] utilizes SAM and image captioning models to predict text descriptions based on the cropped regions, with style adjustments by ChatGPT [60]. Other works train with existing data; GPT4ROI [104] extends Visual LLM [53] to process region prompts, while

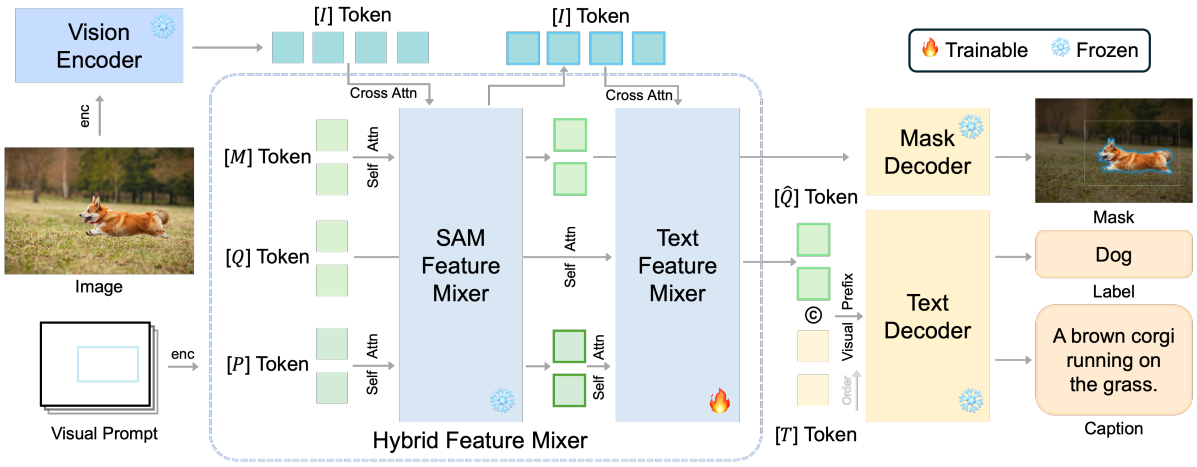

Figure 2. **The model architecture**. The model consists of three parts including an image encoder, a feature mixer, and decoder heads for masks or text. The key ingredient of the model is the *text feature mixer*, which is a lightweight bidirectional transformer [82]. We stack it over the one from SAM and reuse its tokens. By solely optimizing the additional mixer, we align the region-specific features with the embedding space of language models. The training is both fast and scalable thanks to the limited amount of optimizable parameters.

Region-BLIP [107] employs BLIP2's feature mixer [43] and trains on multiple tasks. Works like Kosmos-2 [63] and All-Seeing [87] utilize similar architectures but different dataset construction paths, demonstrating strong performance on various region-level tasks. Despite the rapid evolution of this field, this work aims to extend SAM for region-level captioning with weak supervision.

3. Method

There are three components in the model, a ViT-based encoder, a transformer query-based feature mixer, and decoder heads for different outputs of interest, *e.g.* text decoder. Our model design is inspired by [35], which is a category-agnostic promptable segmentation model that takes in user inputs like points, boxes, or masks and outputs multiple binary masks. Apart from a ViT-based encoder [18, 47] and a small mask decoder [11, 12], it involves a lightweight query-based feature mixer [8] to mix both global image features extracted by the image encoder and the user prompts. The module is efficient as it only consists of 2M of parameters. Fig. 2 illustrate the model architecture.

The data used to train the SAM model is category agnostic and after initial human labeling, the data are scaled to 10M images and 1B boxes with several rounds of self-training. Although initially, the labeled masks involve no textual labels, they contain the semantics **implicitly** as the annotators are asked to draw masks to whatever things or stuff they recognize. Thus we hypothesize that the features from the image encoder of SAM contain rich semantic features beyond the lower-level segmentation tasks it is trained on. Based on that assumption, we build our model over the pre-trained SAM models and stack an additional feature mixer along with a text decoder to predict texts. We follow the mixer design of SAM [35] except for increasing the number of layers in the mixer.

Image encoder. Following SAM, a ViT style image encoder [18] that is designed for detection [47] is adopted in our model. Specifically, it is comprised of a plain ViT with primary local window attention and several interleaved global attention, which produces isotropic feature maps with the same feature dimension.

Given image $\mathcal{I}$, we have the encoder E_I that extract the global image feature I: $E_I(\mathcal{I}) = I$. The image features are down-sampled for computation efficiency, as the following feature mixer should be as lightweight as possible. Following [35, 47], the final spatial shape and feature dimension are 64×64 and 256, respectively (Fig. 2).

Not that we only utilize the single level of visual features from the last layer as in [35], compared with [19, 26, 51, 67, 68] that produce multi-scale features. However, the single-level feature contains sufficient information for later caption generation, regardless of the scales of the regions.

Regional feature mixer. After the global image features are extracted, we need to further extract the region features denoted by the user-input visual prompts. There are two prominent approaches to devising a region feature mixer to attain the region-of-interest (ROI) features. The first one leverages the ROI-align operator [26], which pools the region features from the global ones with the corresponding box coordinates. The second one utilizes the attention mechanism [82] by incorporating query tokens that fuse the feature of interest across each attention block. We choose the latter giving the following considerations: 1) Versatile encoding of visual prompts. The type of visual prompts

could be either point, stroke, box, mask, or a combination of any of them. The ROI-align operator only takes box prompts, while in the query-based token mixer, we can encode the different formats of prompts with specific prompt encoders, whose outputs are tokens that are compatible with the latter attention blocks. 2) Progressive feature interaction and fusion. The main body of the query-based feature mixer is attention blocks as in [8, 12], whose inputs are the encoded prompt tokens, global image tokens, and task-oriented query tokens. After several blocks of self-attentions and cross-attentions, we can fetch the region features at the exact position of the query tokens. Unlike the process of the ROI-align operator, which only pools the global image features, the query-based one can leverage the powerful attention mechanism to extract region-specific features that facilitate the downstream tasks, *e.g*. segmentation, captioning, *etc*.

Given the global image tokens I, and user-provided prompts $\mathcal{P}_{\{b,p,m\}}$ in forms of box b, point p, or mask m, we first encode the given prompts with the corresponding prompt encoders E_p by $E_p(\mathcal{P}_{\{b,p,m\}}) = P_{\{b,p,m\}}$, where $P_{\{b,p,m\}}$ is encoded prompt tokens. Next, we concatenate the encoded prompt tokens and both the textual and mask query tokens Q and M, and feed them with the global image tokens I into the query-based feature mixer E_R with N blocks:

$$E_R^j(P^{j-1}, Q^{j-1}, M^{j-1}; I^{j-1}) = \{\hat{P}^j, \hat{Q}^j, \hat{M}^j; \hat{I}^j\}, \quad (1)$$

where $j = \{1, 2, \ldots, N\}$ is the block indicator, $\{\hat{P}^j, \hat{Q}^j, \hat{M}^j; \hat{I}^j\}$ are the fused tokens after the j-th block, $\{\hat{P}^0, \hat{Q}^0, \hat{M}^0; \hat{I}^0\}$ is the initial input tokens. We denote $\{\hat{P}^N, \hat{Q}^N, \hat{M}^N; \hat{I}^N\} = \{\hat{P}, \hat{Q}, \hat{M}; \hat{I}\}$ as the final outputs. The encoded query tokens $\hat{Q}$ and $\hat{M}$ are deemed as the ROI tokens for captioning and segmentation, respectively, which are delivered to the following output heads (*i.e*., the text generation head and the mask prediction).

The query-based feature mixer E_R is a bi-directional transformer with stack of blocks as in [8, 12, 35, 82]. Each block consists of one self-attention layer to fuse the sparse tokens (*i.e*., the concatenated tokens of the prompt ones P and the query ones Q), and a cross-attention layer to instill the global image tokens I. During the encoding process across each block, the query tokens Q can gradually gather the task-specific information grounded by the prompts ones P, inside the global image tokens I.

Query tokens. [35] takes query-based feature mixer as its core component but only predicts the masks without high-level semantic outputs like labels. We notice that [35] can actually predict masks with good semantics even if it is trained by a category-agnostic approach. It may be attributed to the initial training data of SAM, which are labeled under the instruction where the annotators are asked to draw the masks over whatever things of stuff they recognized without any semantic labels. Thus we leverage the query tokens from [35] by stacking an additional feature mixer E_R^{Cap} above that of [35]. Specifically, [35] possessed its own query tokens M to mix the features for mask prediction. It encoded the corresponding features with a two-layer feature mixer E_R^{SAM}. We add a new set of query tokens Q for text predictions and feed it with the prompt tokens and image tokens that are both encoded with E_R^{SAM} into E_R^{Cap}.

Regional feature decoder. After obtaining the ROI feature, we can send it into a causal text decoder [7, 64, 80, 81] to generate region captions. The text decoder D_{Cap} is often a transformer decoder [82] that predict the text tokens $\mathcal{T}_k$ based on the previous (predicted) text tokens $\mathcal{T}_{1:k-1}$ causally:

$$D_{\text{Cap}}(\mathcal{T}_{1:k-1}) = \mathcal{T}_k, \quad (2)$$

where k is the length of the text tokens. Since we want to condition the prediction on the region features, we prefix the feature token Q in front of the text tokens $\mathcal{T}$. Inspired by prompt tuning [32, 46, 106], we further prefix a set of optimizable task tokens T to exploit task related context (Fig. 2). The model can be optimized by minimizing cross-entropy loss L defined on the next token by:

$$L = \frac{1}{N_{\mathcal{T}}+1} \sum_{k=1}^{N_{\mathcal{T}}+1} \text{CE}(\mathcal{T}_k, p(\mathcal{T}_k|T, Q, \mathcal{T}_{0:k-1})), \quad (3)$$

where $p(\mathcal{T}_k|T, Q, \mathcal{T}_{1:k-1})$ is the predicted logits for token $\mathcal{T}_k$ $N_{\mathcal{T}}$ is the length until the predicted tokens and N_r is the length of the prefix tokens. CE is cross entropy loss with label smoothing at a strength of 0.1. $\mathcal{T}_0, \mathcal{T}_{N_{\mathcal{T}}+1}$ are the begin-of-sentence (BOS) and end-of-sentence (EOS) tokens, respectively. For the details of the mask decoder, please refer to [35].

4. Experiments

4.1. Implementation Details

Our model is comprised of three parts: image encoder, regional feature mixer, and regional feature decoder. The image encoder is the pre-trained ViT-base or -large from [35]. The mask feature mixer along with mask query tokens M and the mask decoder are from the pre-trained SAM. For the text decoder, we leverage the pre-trained language model such as GPT2-large [64] and OpenLLAMA-3B [14, 20, 80]. The above modules are all *fixed* during training. As to the additional transformer region feature mixer to extract textual features, we scale the 2-layer one in [35] to 12 layers. The caption query tokens Q have a length of 8 and the task tokens T have a length of 6. We *optimize* the above modules for region captioning generation. Note that only a small set of parameters are optimized, thus the training is not only scalable but efficient. We list the hyper-parameters in supplementary. We first pre-train the model for 100K steps,

Table 1. Comparison with baselines. "C": CIDEr-D [83], "M": METEOR [5], "S": SPICE [2], "B": BLEU [61], "R": ROUGE [49], "(F)": Fuzzy. For all metrics, the higher the better. The best, the second best, the third best scores are marked as red, orange, yellow, respectively. ∗: The captioners used in [86]. †: We pre-train the model for 100K steps, then finetune it on VG for 100K steps. ‡: When no pertaining is applied, we train the model on VG for 200K steps. Thus they have similar training costs.

Method	C	M	S	B@1	B@2	B@3	B@4	R	Noun	Verb	Noun (F)	Verb (F)
SAM+BLIP-base	43.8	9.6	12.6	16.8	7.8	3.9	2.1	19.8	21.4	3.0	49.6	8.2
SAM+BLIP-large*	25.3	11.0	12.7	14.1	6.5	3.2	1.6	18.5	27.3	4.3	56.2	12.4
SAM+GIT-base	65.5	10.1	17.1	23.6	11.7	7.1	4.8	21.8	22.7	1.4	49.8	3.0
SAM+GIT-base-coco	67.4	11.2	17.5	24.4	12.6	7.5	4.9	23.1	25.6	2.5	52.7	5.2
SAM+GIT-base-textcaps	45.6	11.6	15.0	18.4	8.9	4.7	2.7	21.8	26.1	3.5	54.2	7.4
SAM+GIT-large*	68.8	10.5	17.8	24.2	12.3	7.4	5.0	22.4	24.5	1.8	51.6	3.7
SAM+GIT-large-coco	71.8	12.2	18.8	24.6	12.9	7.7	4.9	24.4	28.9	3.4	55.8	6.7
SAM+GIT-large-textcaps	59.2	12.6	17.5	20.9	10.5	6.0	3.6	23.6	29.4	3.7	56.5	7.2
SAM+BLIP2-OPT-2.7B-coco	30.4	11.3	12.0	14.4	7.1	3.6	1.9	19.3	26.7	4.7	55.0	12.1
SAM+BLIP2-OPT-2.7B*	59.7	11.7	16.7	19.6	9.8	5.3	3.0	22.7	26.6	4.5	53.7	9.7
SAM+BLIP2-OPT-6.7B-coco	30.4	12.2	13.1	14.7	7.3	3.8	2.0	19.9	29.7	4.7	57.8	11.7
SAM+BLIP2-OPT-6.7B	56.6	11.7	16.2	19.0	9.5	5.0	2.8	22.3	26.7	4.4	53.9	10.1
GRiT	142.2	17.2	30.5	36.0	22.1	15.2	11.2	34.5	39.5	4.3	63.3	7.2
SCA (GPT2-large, VG)‡	148.8	17.4	31.2	38.0	23.9	16.6	12.1	35.5	41.5	4.8	65.0	7.6
SCA (LLAMA-3B, VG)‡	149.8	17.4	31.3	38.0	23.9	16.7	12.2	35.5	41.2	4.5	64.6	7.1
SCA (GPT2-large, Pretrain+VG)†	149.8	17.5	31.4	38.2	24.1	16.8	12.2	35.7	41.7	4.8	65.1	7.5

with Objects365 [72] (detection) and COCO-Panoptic [50] (segmentation) with a sampling ratio of 10:1. Then we finetune the model on Visual Genome [36] dense caption split for another 100K steps. Meanwhile, we also directly train the models on VG for 200K steps. For inference, we use a beam size of 3 for text generation. Note that as only the lightweight text feature mixer is optimized, we can *switch* it during inference to generate either class labels (from pertaining) or captions (from finetuning). We list more details in the supplementary materials.

4.2. Evaluation Settings

Datasets. We evaluate the methods on Visual Genome (VG) [36] captioning splits. It contains about 100K images along with around 3M regions, and each region contains one textual description. Despite the large scale of regions, there are a large number of repeated annotations due to its data curation. We take the standard data split protocol [33, 89, 91], in which around 70K images are used for training, and other 5K images are used for evaluation. Compared with previous works [33, 89], we do not preprocess the text (e.g., case conversion, remove the symbols, *etc*.), as we find no performance degradation thanks to the employment of pre-trained language models.

Metrics. We adopt the standard referring-based text similarity measurements [2, 5, 49, 61, 83] used in image captioning [43, 84, 96], to evaluate the generated regional captions against the ground-truth ones. Unlike dense captioning task [33, 91] which considers both localization and generation, we assume localization proposals as given inputs and focus on region captioning. Moreover, we evaluate the concepts learned by the models with phrase coverage rate. We parse both sentences into phrases and then compute the coverage score via Intersection Over Union (IoU) for both nouns and verbs [9]. The score for each pair is either exact matching or fuzzy matching, *i.e*. the cosine similarity between the phrase embeddings. Finally, we average the scores across all samples.

4.3. Comparison with other methods

We compare our methods with two kinds of baselines on the test split of VG. The first baseline is *training-free*, which is a SAM followed by an image captioner [42, 43, 84]. It is the major algorithm in Caption Anything [86]. We evaluate various open-sourced captioners; The second baseline is the GRiT model [89], which is trained on the train split of VG like ours. However, it contains a region generator to automatically generate region proposals, while ours requires those from users. We directly test its captioning ability by providing ground truth boxes.

Tab. 1 demonstrates the superior results of our models. The image captioner baselines yield the least performance. We speculate that the image patches generated by SAM lose the context information, and they differ from the training distribution of the captions w.r.t. both resolution and semantics. Thus it could generate captions that are either misinformative or unspecific. The second baseline, GRiT, gives the most competitive results, but it possesses major drawbacks in comparison with ours. 1) The full model of GRiT, including the image encoder, region proposal net, and text decoder head, is optimized during training, which costs a vast amount of training resources. Our model only optimizes the

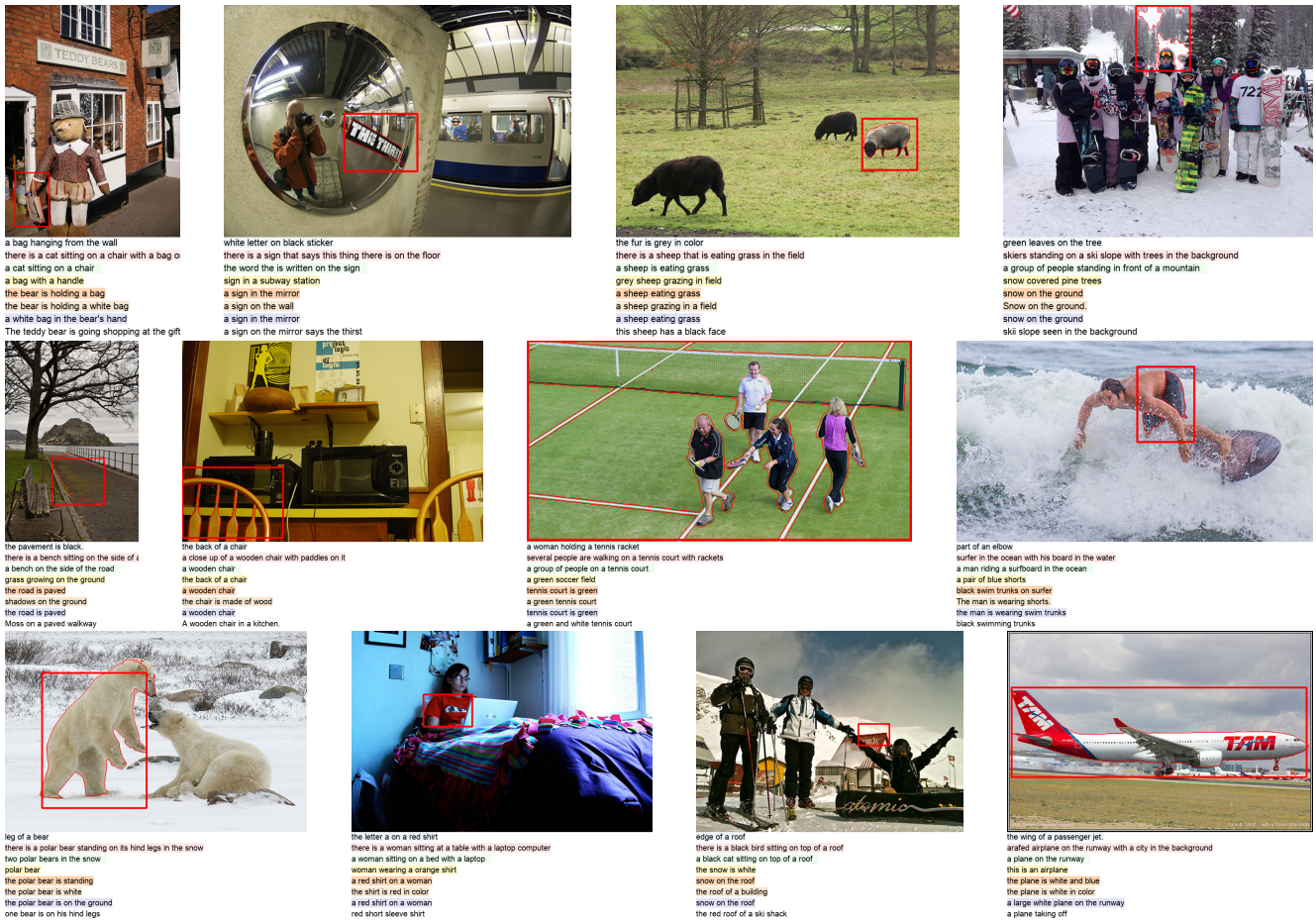

Figure 3. The qualitative results. SCA simultaneously predicts masks (in red contour) and captions. From top-to-bottom, the captions are from: *SAM+Captioner* { GIT-large , BLIP-large , BLIP2-OPT-2.7B } [86], GRIT [89], *SCA* { GPT2-large+VG , LLAMA-3B+VG , GPT2-large+Pretrain+VG }, and the *ground truth*. The bounding boxes (in red) are used to prompt the models. Zoom in for a better view.

lightweight feature mixer, which reduces the cost by lessening the memory consumption and bandwidth for syncing gradients. 2) The text decoding is initialized from scratch in GRiT, which constrains its language modeling ability due to the limited amount of region captioning data. Whereas our method leverages pre-trained language models by mapping SAM's visual features into the language embedding space. It raises two merits: 1) As the power of language model scales, we can observe improved performance on tests. Our model with LLAMA-3B yields superior performance on the VG test set. 2) Since we do not finetune the language model to adapt new data distributions, it is possible to further improve our model based on the language aspect, *e.g.*, chat-style interaction [53, 62]. Fig. 3 visualizes the predictions.

4.4. Ablation Study

In the early stage of experimenting, we spent less computational budgets to validate the efficiency of different design choices. Specifically, for all models in this section, we constrained the budges to 8 16GB V100 GPUs with a batch size of 8. By default, the models are trained solely on VG [36] *without* data augmentation for 200K steps.

Table 2. The ablation of pretraining with weak supervision. $*$: The model is trained solely on VG [36] for *100K* steps. †: The model is first pre-trained for 100K, and then it is fine-tuned for 100K. The training setting for ablations is different from that of Tab. 1.

Pretrain	C	M	S
No Pretrain*	127.9	15.8	27.7
COCO [50] (img. 117K, cls. 80)†	130.2	16.0	28.0
V3Det [85] (img. 183K, cls. 13K)†	130.4	16.0	28.0
O365 [72] (img. 1M, cls. 365)†	134.5	16.3	28.7

The effectiveness of weak supervision pre-training. To preliminarily validate the effectiveness of pretraining with weak supervision. We leveraged three object detection

Table 3. The ablation of training settings of the feature mixer and the text decoder. "M.": Feature mixer, "T.D.": Text decoder.

M. LR	T.D.	T.D. LR	C	M	S
1e-4	GPT2 -large	5e-6	135.6	16.3	28.5
		1e-6	134.8	16.2	28.5
		5e-7	134.5	16.2	28.5
		1e-7	135.6	16.4	28.8
		0.0	136.0	16.5	28.9
5e-5	GPT2 -large	5e-6	129.1	15.7	27.5
		1e-6	131.4	15.9	28.0
		5e-7	131.2	16.0	28.0
		1e-7	132.5	16.1	28.2
		0.0	131.7	16.1	28.2
1e-4	GPT2	5e-6	134.1	16.2	28.4
		1e-6	134.7	16.3	28.7
		5e-7	134.5	16.2	28.7
		1e-7	133.2	16.1	28.6
		0.0	132.3	15.9	28.9
5e-5	GPT2	5e-6	131.3	16.0	28.0
		1e-6	131.1	16.0	28.1
		5e-7	130.6	15.9	28.1
		1e-7	130.4	15.9	28.2
		0.0	126.3	15.4	27.9

datasets: 1) MS COCO [50] contains about 117K images and 80 classes; 2) V3Det [85] is a rich-semantic detection dataset with around 183K images and 13k classes; 3) Objects365 [72] is a large-scale detection dataset with over 1M images, 27M regions, and 365 class labels. The model was first pre-trained for 100K steps and finetuned for another 100K without other modifications. We set another baseline trained directly on VG for 100K steps. Tab. 2 presents that the pretraining with concept labels can facilitate the convergence of training on VG, and the larger the scale of the images, the better the test performance. Under a similar amount of samples, an increase in class labels can slightly improve performance. The finding encourages us to further enlarge the pretraining data scale in the future.

The hyper-parameters of the text decoder and the feature mixer. To determine the training recipe for the text decoder, we experimented with two factors: 1) The size of the text decoder; and 2) The optimization of the text decoder. We tried two variants of GPT2 transformer decoder models [64], which are GPT2-large with 774M parameters and GPT2 with 127M. They are all from the official release which are trained on WebText dataset [64]. Given the different learning rates of the feature mixer, we then tested different learning rates (*i.e.*, 0.0, 1e-7, 5e-7, 1e-6, 5e-6) for the text decoder.

Two conclusions can be drawn from Tab. 3. 1) The feature mixer requires a relatively large learning rate to converge to good performance. 2) When the text decoder is small (GPT2 with 127M), we need to finetune it to achieve better results. In contrast, using a larger text decoder like GPT2-large (774M), finetuning may impede the performance, and fixing the decoder can yield even better scores compared with the small one.

We chose a large text decoder without any finetuning in this paper given the considerations of both capacity and efficiency. In this, we not only keep the knowledge inside the language model for future improvement, but enable low-cost training of the model.

Table 4. The effect of different number of layers in the feature mixer. Note that this is the *only* trainable module in our models.

# of Layers	# of Params	C	M	S
2	3.3 M	108.8	13.6	24.6
4	6.5 M	109.8	14.0	25.6
8	12.8 M	127.0	15.3	27.8
12	19.1 M	127.7	15.3	27.9
24	38.0 M	124.5	15.0	27.3

The size of feature mixer. The additional feature mixer for text decoding is a bi-directional transformer, which fuses the query and prompt tokens with self-attention, and the image tokens with cross-attention. The original one used in [35] is a two-layer one that is highly computational-efficient with solely 3M parameters.

To investigate how the size of the feature mixer affects the extracted region feature, we test a different number of layers for the additional transformer, ranging from 2 with 3M parameters to 24 with 30M parameters.

Tab. 4 demonstrates the final scores on the VG test split. As the number of layers increases, the n-gram metrics ramp up as well. Only until 12 layers do its performance reach the peak, then adding more layers harms the performance. Noticeably, [43] used 12 layers feature mixer to extract prominent features for image captioning, which has over 105M parameters. While ours only consists of 19.4M.

The architecture of feature mixer. We experiment with four major architectures, which are 1) the one with ROI-Align operator [26], which is described in the main paper; 2) the one that directly decoding the query tokens from SAM; 3) the one that does not rely on the fused tokens from SAM's feature mixer (in other words, not reusing SAM's tokens); 4) the one that utilizes the query tokens from SAM to decode texts. To make the ROI align one stronger, we add an MLP stated in [52], which is a two-layer MLP with GELU activation [28].

Tab. 5 shows that query-based mixers perform significantly better than those using ROI-align, indicating the effectiveness of progressive feature aggregation. Directly decoding SAM's query token restricts the capacity of the mixer. Incorporating additional query tokens for captioning

Table 5. The ablation of feature mixer design.

Method	C	M	S
ROI Align [26]	45.2	9.4	11.6
ROI Align + MLP [52]	82.5	12.1	19.3
SAM Query [35]	130.6	15.9	28.4
Text Query w/o SAM Tokens	136.6	16.4	29.2
Text Query w/ SAM Tokens	137.4	16.5	29.3

Table 6. The ablation of using different sizes of image encoder.

Method	# of Params	C	M	S
SAM-ViT-base	86M	130.2	16.0	28.2
SAM-ViT-large	307M	129.6	15.9	28.3
SAM-ViT-huge	632M	130.9	16.0	28.5

boosts the performance of the model. Moreover, resuing the features of SAM further improves the captioning results.

The size of the SAM image encoder. We investigate how different SAM encoders may affect the captioning performance, by testing the three official encoders from [35], which are three ViT [18] with different scale: base, large, and huge. Surprisingly, different size of the SAM image encoders results in similar final performance. We chose the ViT huge as the default image encoder as it performs slightly better among others.

The efficacy of data augmentation. We found that with an enlarged batch size in multiple-node training, the model experienced an overfitting problem which led to inferior test performance. To fight against the problem, we resort to strong augmentation from [21], the large-scale jittering. Tab. 7 demonstrates that using the strong augmentation not only alleviates the overfitting problem but enhances the model's performance.

5. Conclusions and Discussions

We preliminarily demonstrate a regional captioning system by adapting a powerful class-agnostic segmentation model, SAM [35], with a lightweight (typically in the order of tens of millions) query-based feature mixer that bridges SAM with the language model. The mixer is the only optimizable module thus the training is both *faster* and *scalable*, as it costs less computation, less memory usage, and less communication bandwidth. To better generalize our model, we pre-train the system with weak supervision which transfers the general knowledge of the visual concepts beyond the limited regional captioning data, Visual Genome (VG) [36]. We extensively validate our design choices and evaluate our method, demonstrating its strong performance.

Limitations. 1) Wrong attribute prediction. *e.g.*, the models could predict the wrong colors or textures; 2) Distinguishing similar visual concepts. *e.g.*, the model may confuse "lemon" with "orange"; 3) Alignment with mask predictions: As we do not supervise the alignment, the model may predict mask and captions for the fore- and background separately. The drawbacks, *esp.* 1) and 2), may be addressed by weak supervision and self-training [6].

Table 7. The ablation of using data augmentation. "LM": Language model, "Aug.": Augmentation.

LM	Aug.	C	M	S
GPT2-large	No LSJ	137.6	16.5	29.3
	LSJ (1.0, 2.0)	140.2	16.7	29.9
	LSJ (0.1, 2.0)	140.8	16.7	29.9
LLAMA-3B	No LSJ	137.7	16.4	29.2
	LSJ (1.0, 2.0)	142.1	16.7	30.0
	LSJ (0.1, 2.0)	142.6	16.8	30.1

Weak supervision and self-training. We only leverage 1.8M weak supervision data [50, 72] to complement the regional captioning data, VG [36]. Our ablation about the effectiveness of weak supervision shows that the scale of images matters more than the variety of labels, which is intuitive as we want the model to align and *generalize* as much as visual concepts with the language models. Thus pertaining the model with *bigger datasets* like [38] may lead to better generalizability. Another approach to leverage *image captioning* data as in [44, 111], but it requires to solve the problem of granularity mismatching [111]. *Self-training* is the ultimate goal that could scale both the data and the generalizability of the model. It demonstrates effectiveness in image captioning [42], segmentation [35], open-vocabulary detection [57], and text-to-image generation [6]. We believe this work serves as a footstone towards scaling regional captioning data [6, 35, 57] in the future.

Insight of lifting SAM for regional captioning. Although there are no semantic labels in the training data, SAM still implies *high-level semantics* that are sufficient for captioning. The masks used to train SAM are labeled in a way where annotators are asked to draw masks for every *things* or *stuff* they recognized [35]. After several rounds of self-training and bootstrapping the data to 1B masks, the attained models possess implicit general knowledge about the visual world. Therefore, we can *align* the implicit general knowledge with natural languages to caption regions. We believe this work sheds light on exploring the emerging ability [88] in vision from low-level data or pre-trains.

Acknowledgments. This work was supported in part by the National Natural Science Foundation of China under Grant 62125603, Grant 62321005, and Grant 62336004.

References

[1] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob L. Menick, Sebastian Borgeaud, Andy Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karén Simonyan. Flamingo: A visual language model for few-shot learning. In *NeurIPS*, 2022. 2

[2] Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. Spice: Semantic propositional image caption evaluation. In *ECCV*, pages 382–398, 2016. 5

[3] Rohan Anil, Andrew M. Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, Eric Chu, Jonathan H. Clark, Laurent El Shafey, Yanping Huang, Kathy Meier-Hellstern, Gaurav Mishra, Erica Moreira, Mark Omernick, Kevin Robinson, Sebastian Ruder, Yi Tay, Kefan Xiao, Yuanzhong Xu, Yujing Zhang, Gustavo Hernández Ábrego, Junwhan Ahn, Jacob Austin, Paul Barham, Jan A. Botha, James Bradbury, Siddhartha Brahma, Kevin Brooks, Michele Catasta, Yong Cheng, Colin Cherry, Christopher A. Choquette-Choo, Aakanksha Chowdhery, Clément Crepy, Shachi Dave, Mostafa Dehghani, Sunipa Dev, Jacob Devlin, Mark Díaz, Nan Du, Ethan Dyer, Vladimir Feinberg, Fangxiaoyu Feng, Vlad Fienber, Markus Freitag, Xavier Garcia, Sebastian Gehrmann, Lucas Gonzalez, and et al. PaLM 2 technical report. *arXiv preprint arXiv:2305.10403*, 2023. 2

[4] Relja Arandjelović, Alex Andonian, Arthur Mensch, Olivier J. Hénaff, Jean-Baptiste Alayrac, and Andrew Zisserman. Three ways to improve feature alignment for open vocabulary detection, 2023. 2

[5] Satanjeev Banerjee and Alon Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *ACL Workshop*, pages 65–72, 2005. 5

[6] James Betker, Gabriel Goh, Li Jing, Tim Brooks, Jianfeng Wang, Linjie Li, Long Ouyang, Juntang Zhuang, Joyce Lee, Yufei Guo, Wesam Manassra, Prafulla Dhariwal, Casey Chu, Yunxin Jiao, and Aditya Ramesh. Improving Image Generation with Better Captions. *OpenAI blog*, 2023. 2, 8

[7] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *NeurIPS*, 2020. 1, 2, 4

[8] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020. 2, 3, 4

[9] David M Chan, Austin Myers, Sudheendra Vijayanarasimhan, David A Ross, and John Canny. Ic3: Image captioning by committee consensus. *arXiv preprint arXiv:2302.01328*, 2023. 5

[10] Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12M: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *CVPR*, 2021. 2

[11] Bowen Cheng, Alexander G. Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. In *NeurIPS*, 2021. 2, 3

[12] Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *CVPR*, 2022. 2, 3, 4

[13] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. PaLM: Scaling language modeling with pathways. *JMLR*, 2023. 2

[14] Together Computer. RedPajama-Data: An Open Source Recipe to Reproduce LLaMA training dataset, 2023. 4

[15] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven C. H. Hoi. InstructBLIP: Towards general-purpose vision-language models with instruction tuning. *arXiv preprint arXiv:2305.06500*, 2023. 2

[16] Mingyu Ding, Bin Xiao, Noel Codella, Ping Luo, Jingdong Wang, and Lu Yuan. DaViT: Dual attention vision transformers. In *ECCV*, 2022. 2

[17] Xiaoyi Dong, Jianmin Bao, Yinglin Zheng, Ting Zhang, Dongdong Chen, Hao Yang, Ming Zeng, Weiming Zhang, Lu Yuan, Dong Chen, Fang Wen, and Nenghai Yu. MaskCLIP: Masked self-distillation advances contrastive language-image pretraining. In *CVPR*, 2023. 2

[18] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An im-

age is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. 2, 3, 8

[19] Kaiwen Duan, Song Bai, Lingxi Xie, Honggang Qi, Qingming Huang, and Qi Tian. CenterNet: Keypoint triplets for object detection. In *ICCV*, 2019. 2, 3

[20] Xinyang Geng and Hao Liu. OpenLLaMA: An Open Reproduction of LLaMA, 2023. 4

[21] Golnaz Ghiasi, Yin Cui, Aravind Srinivas, Rui Qian, Tsung-Yi Lin, Ekin D. Cubuk, Quoc V. Le, and Barret Zoph. Simple copy-paste is a strong data augmentation method for instance segmentation. In *CVPR*, 2021. 8

[22] Golnaz Ghiasi, Xiuye Gu, Yin Cui, and Tsung-Yi Lin. Scaling open-vocabulary image segmentation with image-level labels. In *ECCV*, 2022. 2

[23] Ross B. Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014. 2

[24] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016. 1

[25] L. Grady. Random Walks for Image Segmentation. *TPAMI*, 28(11):1768–1783, 2006. 2

[26] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask R-CNN. *TPMAI*, 2020. 2, 3, 7, 8

[27] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross B. Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, 2022. 2

[28] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016. 7

[29] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and Laurent Sifre. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022. 2

[30] MD Zakir Hossain, Ferdous Sohel, Mohd Fairuz Shiratuddin, and Hamid Laga. A comprehensive survey of deep learning for image captioning. *ACMCS*, 51(6):1–36, 2019. 1

[31] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V. Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *ICML*, 2021. 1, 2

[32] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge J. Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *ECCV*, 2022. 4

[33] Justin Johnson, Andrej Karpathy, and Li Fei-Fei. DenseCap: Fully convolutional localization networks for dense captioning. In *CVPR*, 2016. 1, 2, 5

[34] Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. *TPMAI*, 2017. 1, 2

[35] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloé Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross B. Girshick. Segment anything. *arXiv preprint arXiv:2304.02643*, 2023. 2, 3, 4, 7, 8

[36] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Li Fei-Fei. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *IJCV*, 2017. 2, 5, 6, 8

[37] Girish Kulkarni, Visruth Premraj, Vicente Ordonez, Sagnik Dhar, Siming Li, Yejin Choi, Alexander C Berg, and Tamara L Berg. Babytalk: Understanding and generating simple image descriptions. *TPAMI*, 35(12):2891–2903, 2013. 1

[38] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper R. R. Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Tom Duerig, and Vittorio Ferrari. The Open Images Dataset V4: Unified image classification, object detection, and visual relationship detection at scale. *arXiv preprint arXiv:1811.00982*, 2018. 2, 8

[39] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015. 1

[40] Boyi Li, Kilian Q. Weinberger, Serge J. Belongie, Vladlen Koltun, and René Ranftl. Language-driven semantic segmentation. In *ICLR*, 2022. 2

[41] Feng Li, Hao Zhang, Shilong Liu, Jian Guo, Lionel M. Ni, and Lei Zhang. DN-DETR: Accelerate DETR training by introducing query DeNoising. In *CVPR*, 2022. 2

[42] Junnan Li, Dongxu Li, Caiming Xiong, and Steven C. H. Hoi. BLIP: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*, 2022. 1, 2, 5, 8

[43] Junnan Li, Dongxu Li, Silvio Savarese, and Steven C. H. Hoi. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *ICML*, 2023. 1, 2, 3, 5, 7

[44] Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, Kai-Wei Chang, and Jianfeng Gao. Grounded language-image pre-training. In *CVPR*, 2022. 2, 8

[45] Xiangyang Li, Shuqiang Jiang, and Jungong Han. Learning object context for dense captioning. In *AAAI*, pages 8650–8657, 2019. 2

[46] Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. In *ACL/IJCNLP*, 2021. 4

[47] Yanghao Li, Hanzi Mao, Ross B. Girshick, and Kaiming He. Exploring plain vision transformer backbones for object detection. In *ECCV*, 2022. 3

[48] Feng Liang, Bichen Wu, Xiaoliang Dai, Kunpeng Li, Yinan Zhao, Hang Zhang, Peizhao Zhang, Peter Vajda, and Diana Marculescu. Open-vocabulary semantic segmentation with mask-adapted CLIP. In *CVPR*, 2023. 2

[49] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81, 2004. 5

[50] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. In *ECCV*, 2014. 2, 5, 6, 7, 8

[51] Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. *TPMAI*, 42(2):318–327, 2020. 3

[52] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. *arXiv preprint arXiv:2310.03744*, 2023. 7, 8

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

[54] Shilong Liu, Feng Li, Hao Zhang, Xiao Yang, Xianbiao Qi, Hang Su, Jun Zhu, and Lei Zhang. DAB-DETR: Dynamic anchor boxes are better queries for DETR. In *ICLR*, 2022. 2

[55] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott E. Reed, Cheng-Yang Fu, and Alexander C. Berg. SSD: Single shot MultiBox detector. In *ECCV*, 2016. 2

[56] Yanxin Long, Youpeng Wen, Jianhua Han, Hang Xu, Pengzhen Ren, Wei Zhang, Shen Zhao, and Xiaodan Liang. CapDet: Unifying Dense Captioning and Open-World Detection Pretraining. In *CVPR*, pages 15233–15243, 2023. 1, 2

[57] Matthias Minderer, Alexey A. Gritsenko, and Neil Houlsby. Scaling open-vocabulary object detection. *arXiv preprint arXiv:2306.09683*, 2023. 2, 8

[58] OpenAI. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 2

[59] Vicente Ordonez, Girish Kulkarni, and Tamara L. Berg. Im2Text: Describing images using 1 million captioned photographs. In *NeurIPS*, 2011. 2

[60] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In *NeurIPS*, 2022. 2

[61] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *ACL*, pages 311–318, 2002. 5

[62] Baolin Peng, Chunyuan Li, Pengcheng He, Michel Galley, and Jianfeng Gao. Instruction tuning with GPT-4. *arXiv preprint arXiv:2304.03277*, 2023. 1, 2, 6

[63] Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, and Furu Wei. Kosmos-2: Grounding multimodal large language models to the world. *arXiv preprint arXiv:2306.14824*, 2023. 3

[64] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language Models are Unsupervised Multitask Learners. *OpenAI blog*, 1(8):9, 2019. 2, 4, 7

[65] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 1, 2

[66] Kate Rakelly, Evan Shelhamer, Trevor Darrell, Alexei A. Efros, and Sergey Levine. Few-shot segmentation propagation with guided networks. *CoRR*, abs/1806.07373, 2018. 2

[67] Joseph Redmon, Santosh Kumar Divvala, Ross B. Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *CVPR*, 2016. 2, 3

[68] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. *TPMAI*, 2017. 2, 3

[69] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, pages 10684–10695, 2022. 2

[70] Carsten Rother, Vladimir Kolmogorov, and Andrew Blake. "GrabCut": Interactive foreground extraction using iterated graph cuts. *TOG*, 23(3):309–314, 2004. 2

[71] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *NeurIPS*, 35:36479–36494, 2022. 2

[72] Shuai Shao, Zeming Li, Tianyuan Zhang, Chao Peng, Gang Yu, Xiangyu Zhang, Jing Li, and Jian Sun. Objects365: A large-scale, high-quality dataset for object detection. In *ICCV*, 2019. 2, 5, 6, 7, 8

[73] Zhuang Shao, Jungong Han, Demetris Marnerides, and Kurt Debattista. Region-Object Relation-Aware Dense Captioning via Transformer. *TNNLS*, pages 1–12, 2022. 2

[74] Himanshu Sharma, Manmohan Agrahari, Sujeet Kumar Singh, Mohd Firoj, and Ravi Kumar Mishra. Image captioning: a comprehensive survey. In *PARC*, pages 325–328, 2020. 1

[75] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *ACL*, 2018. 2

[76] Matteo Stefanini, Marcella Cornia, Lorenzo Baraldi, Silvia Cascianelli, Giuseppe Fiameni, and Rita Cucchiara. From show to tell: A survey on deep learning-based image captioning. *TPAMI*, 45(1):539–559, 2022. 1

[77] Quan Sun, Yuxin Fang, Ledell Wu, Xinlong Wang, and Yue Cao. EVA-CLIP: Improved training techniques for CLIP at scale. *arXiv preprint arXiv:2303.15389*, 2023. 2

[78] Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Stanford alpaca: An instruction-following llama model. https://github.com/tatsu-lab/stanford_alpaca, 2023. 1

[79] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. FCOS: Fully convolutional one-stage object detection. In *ICCV*, 2019. 2

[80] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurélien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. LLaMA: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. 1, 2, 4

[81] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton-Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurélien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023. 1, 2, 4

[82] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 2, 3, 4

[83] Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In *CVPR*, pages 4566–4575, 2015. 5

[84] Jianfeng Wang, Zhengyuan Yang, Xiaowei Hu, Linjie Li, Kevin Lin, Zhe Gan, Zicheng Liu, Ce Liu, and Lijuan Wang. GIT: A generative image-to-text transformer for vision and language. *TMLR*, 2022. 1, 2, 5

[85] Jiaqi Wang, Pan Zhang, Tao Chu, Yuhang Cao, Yujie Zhou, Tong Wu, Bin Wang, Conghui He, and Dahua Lin. V3Det: Vast vocabulary visual detection dataset. *arXiv preprint arXiv:2304.03752*, 2023. 2, 6, 7

[86] Teng Wang, Jinrui Zhang, Junjie Fei, Hao Zheng, Yunlong Tang, Zhe Li, Mingqi Gao, and Shanshan Zhao. Caption anything: Interactive image description with diverse multimodal controls. *arXiv preprint arXiv:2305.02677*, 2023. 2, 5, 6

[87] Weiyun Wang, Min Shi, Qingyun Li, Wenhai Wang, Zhenhang Huang, Linjie Xing, Zhe Chen, Hao Li, Xizhou Zhu, Zhiguo Cao, Yushi Chen, Tong Lu, Jifeng Dai, and Yu Qiao. The all-seeing project: Towards panoptic visual recognition and understanding of the open world. *arXiv preprint arXiv:2308.01907*, 2023. 3

[88] Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed Huai hsin Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. Emergent abilities of large language models. *TMLR*, 2022, 2022. 8

[89] Jialian Wu, Jianfeng Wang, Zhengyuan Yang, Zhe Gan, Zicheng Liu, Junsong Yuan, and Lijuan Wang. GRiT: A generative region-to-text transformer for object understanding. *arXiv preprint arXiv:2212.00280*, 2022. 1, 2, 5, 6

[90] Mengde Xu, Zheng Zhang, Fangyun Wei, Han Hu, and Xiang Bai. Side adapter network for open-vocabulary semantic segmentation. In *CVPR*, pages 2945–2954, 2023. 2

[91] Linjie Yang, Kevin D. Tang, Jianchao Yang, and Li-Jia Li. Dense captioning with joint inference and visual context. In *CVPR*, pages 1978–1987, 2017. 2, 5

[92] Lewei Yao, Jianhua Han, Youpeng Wen, Xiaodan Liang, Dan Xu, Wei Zhang, Zhenguo Li, Chunjing Xu, and Hang Xu. DetCLIP: Dictionary-enriched visual-concept paralleled pre-training for open-world detection. In *NeurIPS*, 2022. 2

[93] Lewei Yao, Jianhua Han, Xiaodan Liang, Dan Xu, Wei Zhang, Zhenguo Li, and Hang Xu. DetCLIPv2: Scalable open-vocabulary object detection pre-training via word-region alignment. In *CVPR*, 2023. 2

[94] Muyang Yi, Quan Cui, Hao Wu, Cheng Yang, Osamu Yoshie, and Hongtao Lu. A simple framework for text-supervised semantic segmentation. In *CVPR*, 2023. 2

[95] Guojun Yin, Lu Sheng, Bin Liu, Nenghai Yu, Xiaogang Wang, and Jing Shao. Context and attribute grounded dense captioning. In *CVPR*, pages 6241–6250, 2019. 2

[96] Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint arXiv:2205.01917*, 2022. 1, 5

[97] Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, et al. Scaling autoregressive models for content-rich text-to-image generation. *arXiv preprint arXiv:2206.10789*, 2022. 2

[98] Licheng Yu, Patrick Poirson, Shan Yang, Alexander C. Berg, and Tamara L. Berg. Modeling context in referring expressions. In *ECCV*, 2016. 2

[99] Lu Yuan, Dongdong Chen, Yi-Ling Chen, Noel Codella, Xiyang Dai, Jianfeng Gao, Houdong Hu, Xuedong Huang, Boxin Li, Chunyuan Li, Ce Liu, Mengchen Liu, Zicheng Liu, Yumao Lu, Yu Shi, Lijuan Wang, Jianfeng Wang, Bin Xiao, Zhen Xiao, Jianwei Yang, Michael Zeng, Luowei Zhou, and Pengchuan Zhang. Florence: A new foundation model for computer vision. *arXiv preprint arXiv:2111.11432*, 2021. 1, 2

[100] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. *arXiv preprint arXiv:2303.15343*, 2023. 2

[101] Haotian Zhang, Pengchuan Zhang, Xiaowei Hu, Yen-Chun Chen, Liunian Harold Li, Xiyang Dai, Lijuan Wang, Lu Yuan, Jenq-Neng Hwang, and Jianfeng Gao. GLIPv2: Uni-

fying localization and vision-language understanding. In *NeurIPS*, 2022. 2

[102] Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel M. Ni, and Heung-Yeung Shum. DINO: DETR with improved DeNoising anchor boxes for end-to-end object detection. In *ICLR*, 2023. 2

[103] Hao Zhang, Feng Li, Xueyan Zou, Shilong Liu, Chunyuan Li, Jianwei Yang, and Lei Zhang. A simple framework for open-vocabulary segmentation and detection. In *ICCV*, pages 1020–1031, 2023. 2

[104] Shilong Zhang, Peize Sun, Shoufa Chen, Min Xiao, Wenqi Shao, Wenwei Zhang, Kai Chen, and Ping Luo. GPT4RoI: Instruction tuning large language model on region-of-interest. *arXiv preprint arXiv:2307.03601*, 2023. 1, 2

[105] Yiwu Zhong, Jianwei Yang, Pengchuan Zhang, Chunyuan Li, Noel Codella, Liunian Harold Li, Luowei Zhou, Xiyang Dai, Lu Yuan, Yin Li, and Jianfeng Gao. RegionCLIP: Region-based language-image pretraining. In *CVPR*, 2022. 2

[106] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *IJCV*, 2022. 4

[107] Qiang Zhou, Chaohui Yu, Shaofeng Zhang, Sitong Wu, Zhibing Wang, and Fan Wang. RegionBLIP: A unified multi-modal pre-training framework for holistic and regional comprehension. *arXiv preprint arXiv:2308.02299*, 2023. 3

[108] Xingyi Zhou, Rohit Girdhar, Armand Joulin, Philipp Krähenbühl, and Ishan Misra. Detecting twenty-thousand classes using image-level supervision. In *ECCV*, 2022. 2

[109] 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

[110] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable DETR: Deformable transformers for end-to-end object detection. In *ICLR*, 2021. 2

[111] Xueyan Zou, Zi-Yi Dou, Jianwei Yang, Zhe Gan, Linjie Li, Chunyuan Li, Xiyang Dai, Harkirat Behl, Jianfeng Wang, Lu Yuan, et al. Generalized decoding for pixel, image, and language. In *CVPR*, pages 15116–15127, 2023. 2, 8