UniTAB: Unifying Text and Box Outputs for Grounded Vision-Language Modeling

Zhengyuan Yang, Zhe Gan, Jianfeng Wang, Xiaowei Hu,
Faisal Ahmed, Zicheng Liu, Yumao Lu, Lijuan Wang

Microsoft Cloud and AI
{zhengyang,zhe.gan,jianfw,xiaowei.hu,
fiahmed,zliu,yumaolu,lijuanw}@microsoft.com

Abstract. We propose UniTAB that Unifies Text And Box outputs for grounded vision-language (VL) modeling. Grounded VL tasks such as grounded captioning require the model to generate a text description and align predicted words with object regions. To achieve this, models must generate desired text and box outputs together, and meanwhile indicate the alignments between words and boxes. In contrast to existing solutions that use multiple separate modules for different outputs, UniTAB represents both text and box outputs with a shared token sequence, and introduces a special $<obj>$ token to naturally indicate word-box alignments in the sequence. UniTAB thus could provide a more comprehensive and interpretable image description, by freely grounding generated words to object regions. On grounded captioning, UniTAB presents a simpler solution with a single output head, and significantly outperforms state of the art in both grounding and captioning evaluations. On general VL tasks that have different desired output formats (*i.e.*, text, box, or their combination), UniTAB with a single network achieves better or comparable performance than task-specific state of the art. Experiments cover 7 VL benchmarks, including grounded captioning, visual grounding, image captioning, and visual question answering. Furthermore, UniTAB's unified multi-task network and the task-agnostic output sequence design make the model parameter efficient and generalizable to new tasks.

1 Introduction

Text sequences [11,5] and bounding boxes [39,72] are two representative output formats for image understanding tasks [16,39,11]. Text is well suited for generating image-level predictions, such as describing an image with a sentence [11] or tagging an image with keywords [20], but fails to refer to a dense image region. On the other hand, box could point to any image area [39], but alone has a limited ability to provide semantically-rich descriptions. A natural question is *can we have a single model that unifies text and box outputs, i.e.*, generating both text and box outputs while aligning predicted words with boxes. Unifying these two output formats allows the model to better express its understanding of the image. Taking captioning as an example, such a unified model could ground all

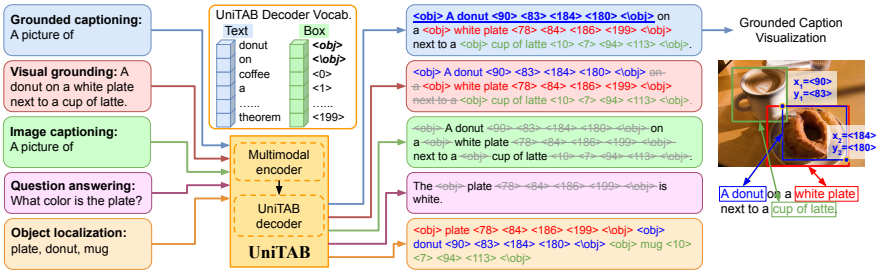

Fig. 1. We propose UniTAB that Unifies Text And Box outputs with no format-specific modules. UniTAB generates both text and box tokens in an auto-regressive manner, conditioned on the multimodal image-text inputs. The introduced *<obj>* token naturally indicates the word-box alignments, as shown in word-box pairs of the same color in the right visualization. UniTAB thus can approach a wide range of VL tasks, including the challenging grounded captioning, with a single unified architecture. The ~~gray~~ tokens in the task-agnostic output sequence are predictions not used for downstream task evaluation, *e.g.*, box tokens in image captioning and VQA.

noun entities [78,49] in the caption back to aligned image regions, thus providing a more comprehensive and interpretable image description. This problem is known as grounded captioning [78,80,45,49]. Moreover, unifying output formats is one important step toward the grand vision of building task-agnostic, general-purpose vision systems [23] that are parameter efficient and well generalizable.

Recent works [13,23,78,80,45] have developed models that can generate both text and box outputs. Specifically, the system combines an online [23] or offline [13,78,80,45] object detection module that predicts boxes, with a vision-language model that generates text. The word and box alignments are then separately generated as additional predictions, such as the relevance score [23,78,80,45]. Predicting text, box, and their alignments separately weakens the benefits of a unified system. The separate modules prevent the framework from being simple and parameter efficient. Furthermore, the explicit object detection component increases the model running time [33] and potentially limits its generalization ability given the preset detector vocabulary [65], as discussed in previous VL studies [33,65]. Going beyond these successful initial explorations, we ask a bolder question: *can we unify the output formats with no separate modules*? Specifically, we explore **1).** how to have a single architecture without an explicit detector jointly generating text and box, and **2).** how to represent the word-box alignments naturally in the output to avoid the additional alignment prediction. To this end, we model both text and box predictions as an auto-regressive token generation task, and present a single encoder-decoder model that is fully shared among text, box, and alignment predictions.

Our modeling of box prediction takes inspiration from Pix2seq [10], an object detection study showing that predicting boxes in an auto-regressive manner yields good detection performance [39]. Its core idea is to quantize the four coordinates in each box into four discrete box tokens, and arrange them with a fixed

order into a token sequence, *i.e.*, $[y_{\min}, x_{\min}, y_{\max}, x_{\max}]$. Box prediction can then be modeled as a multi-step classification task, instead of conventional coordinate regression [22,52,8]. The same classification modeling as in text generation [50] makes it possible to unify text and box prediction. However, Pix2seq is designed for the single-modal object detection task, and does not support open-ended text generation nor multimodal inputs and outputs. Moreover, it is unclear how the text and box alignment is intended to be presented in a unified sequence.

In this study, we propose UniTAB that unifies text and box outputs. As shown in Figure 1, we unify open-ended text generation [50] and discrete box token prediction [10] into a single shared decoder. During the auto-regressive decoding, UniTAB switches to box tokens right after any text words to be grounded, and switches back to text tokens after predicting the box. In UniTAB, we study how to handle such text-box code-switching [67] and naturally represent word-box alignments. We introduce a special <*obj*> token inserted before the text word to be grounded, and after the generated box tokens. The <*obj*> token simplifies the sequence generation by providing hints of the code-switching, and naturally represents word-box alignments. That is, the words and box within a pair of <*obj*> tokens refer to the same entity, as shown in word-box pairs of the same color in Figure 1. With the <*obj*> token and output sequence design, UniTAB approaches grounded VL tasks such as grounded captioning [78,49] and phrase grounding [49] with a single decoder, in contrast to separately predicting text, box, and their alignments with multiple output heads [78,45,80,31].

We further apply UniTAB on general VL tasks [78,72,46,49,5,11,77] and observe two unique properties. *First*, the unified architecture for text, box, and alignment predictions enables UniTAB to perform multi-task training [1,66,6], which learns a single set of parameters for different VL tasks without introducing task-specific heads. Multi-task training avoids task-specific model copies and thus saves the parameters to store. It also facilities the use of data in different tasks, thus boosting the performance of certain VL tasks. *Second*, as shown in Figure 1, UniTAB's output sequence is designed to be task-agnostic and shares the same text+box design across different VL tasks. The task-agnostic output design could help UniTAB generalize to certain unseen tasks, by reformatting new tasks' desired outputs into the seen text+box sequences.

We evaluate UniTAB on 7 VL benchmarks, including grounded captioning [78,49], visual grounding [72,46,49], image captioning [11], and visual question answering [5], all with a single encoder-decoder network architecture, trained by the cross-entropy language modeling objective [50]. With a unified framework and minimum task-specific assumptions, our model achieves better or comparable performance with task-specific state of the art. In grounded captioning, UniTAB not only presents a simpler solution by eliminating separate task-specific heads [78,45,80,31], but also significantly outperforms the prior art [45,9] (from 62.5 to 69.7 in captioning CIDEr score and from 8.44 to 12.95 in grounding F1 score). Our contributions are summarized as follows.

- UniTAB is the first grounded VL model that can approach a wide range of tasks, including the challenging grounded captioning, without separate out-

Table 1. Summary of unified VL models. We highlight the desired modeling in blue. *Visual Modeling:* instead of using an object detection (OD) module, we take raw "image patches" as visual input. *Text Output:* instead of using task-specific output heads [42,29,31,26] for different VL tasks (classification or text generation heads), we use a "single output sequence" [13,23] to approach different tasks. *Box Output:* many prior models cannot predict boxes [29] or simplify it as region index prediction with detector-generated region proposals [42,13,23]. We aim to predict "box coordinates" without an explicit OD module [31,26]. *Word-box Align:* most models fail to generate either open-ended text [42,31,26] or object boxes [29], thus cannot represent word-box alignments. In contrast to the extra alignment predictions [23,13], our introduced <*obj*> token naturally indicates word-box alignments "inline" in the output sequence.

Representative Models	Visual Modeling	Text Output	Box Output	Word-box Align
ViLBERT [42], OSCAR [38], UNITER [12], VinVL [74], *etc.* [37,59,35,58,79,43]	Offline OD	Task-specific Heads	Region Index	✗
PixelBERT [29], SOHO [28], ViLT [33], SimVLM [65], *etc.* [56,36,69,19,63]	Image Patches	Task-specific Heads	✗	✗
VL-T5 [13] GPV [23]	Offline OD Online OD	Single Output Seq.	Region Index	Extra Prediction
MDERT [31], UniT [26]	Image Patches	Task-specific Heads	Box Coordinate	✗
UniTAB (Ours)	Image Patches	Single Output Seq.	Box Coordinate	Inline Indicated

put modules. We introduce the <*obj*> token that helps text and box outputs synergistically work together, with their alignments naturally represented.
– UniTAB achieves better or comparable performance to state of the art on 7 VL benchmarks. Its unified multi-task network and the task-agnostic output sequence design make it parameter efficient and generalizable to new tasks.

2 Related Work

Grounded captioning. The grounded captioning task [78,49] requires the model to generate a text caption and grounds all mentioned noun phrases [78,49] to aligned image regions. The input is a single image, and the desired outputs are the caption sentence, multiple object boxes, and the word-box alignments. Existing methods [78,45,80,9] adopt separate output heads for text, box (usually with an offline detector [53,4]), and alignment predictions. In contrast, UniTAB uses a single decoding sequence to represent all desired outputs.

Vision-language pre-training (VLP). Large-scale VLP has become the new training paradigm for VL research. Prior works [42,37,2,35,59,58,79,12,43,38] first show the power of VLP by using region features obtained from an off-the-shelf object detector [53]. However, the region feature extraction significantly increases the model's computation cost and run time. Recent studies [29,33,36,65] shift the paradigm and show that grid features extracted from raw image patches also work well. Most studies adopt similar output architectures of either discriminative classification heads or auto-regressive text decoders. As shown in the second row of Table 1, these output structures often contain task-specific designs

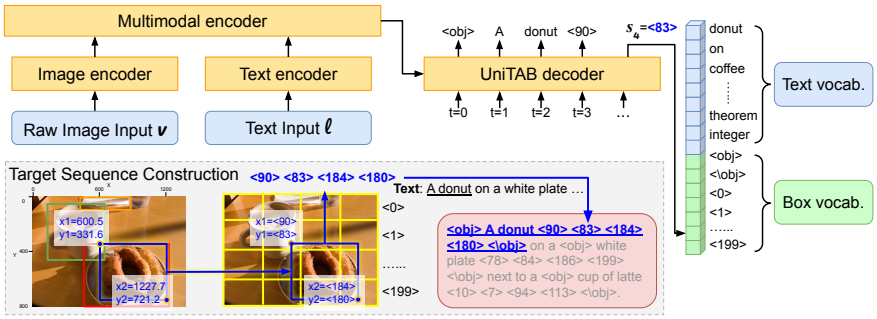

Fig. 2. UniTAB is an encoder-decoder framework that can jointly output open-ended text and box without output format specific modules. A transformer encoder-decoder takes the encoded image-text features to predict the target text+box sequence. The bottom sub-figure illustrates the output target sequence design. We introduce a special *<obj>* token to indicate the alignments between predicted words and boxes, such as words "a donut" and the blue box. During decoding, the output sequence could seamlessly switch between text and box tokens to ground an object, if applicable.

and do not support bounding box prediction, which is an important output format for VL tasks such as visual grounding and grounded image captioning.

Unified VL framework. Prior works have presented successful explorations on building VL models with unified input-output formats. VL-T5 [13] and GPV [23] first represent images as object region features with an online or offline object detector [53,8]. Bounding box prediction is then simplified as index classification over the set of region candidates generated by the detector. The other threads, MDETR [31] and UniT [26], add task-specific classification heads on top of the DETR object detector [8] to perform VL tasks. However, different tasks still require different output heads. Moreover, it is unclear how to extend the framework for open-ended text generation, thus supporting VL tasks like image captioning. In this study, we aim to build a single unified framework that takes structured inputs (*i.e.*, raw image and language) in, and generates structured outputs (*i.e.*, text and boxes), with no output format specific modules.

3 The UniTAB Framework

3.1 Architecture Overview

We implement UniTAB using a transformer encoder-decoder architecture built on top of the single-modality image and text encoders, as shown in Figure 2. For image, we use ResNet-101 [24] to encode the raw image input v, and flatten the grid features as the visual representation. For text, we use $\text{RoBERTa}_{\text{BASE}}$ [40] to encode input text l into hidden word features. The encoded image and text features are then projected into a shared embedding space. We use a 6-layer transformer encoder that takes the concatenated image and text feature sequence

as input, and a 6-layer transformer decoder for output sequence generation. The decoder generates output tokens in an auto-regressive manner, similar to language modeling [50,51]. The UniTAB decoder could generate tokens from both the text and box vocabularies, as shown in the right part of Figure 2.

3.2 UniTAB Target Output Sequence

We show how to construct ground-truth target output sequences, such that text and box can be jointly represented with word-box alignments contained inline.

Box token sequence. We first review the bounding box quantization approach introduced in Pix2seq [10]. As shown in the bottom part of Figure 2, a rectangular bounding box in a 2D image can be represented by four floating-point numbers, namely $[x_{\text{min}}, y_{\text{min}}, x_{\text{max}}, y_{\text{max}}]$. The established object detection paradigm [53,52,8] predicts four continuous floating-point values to regress the coordinates in a single step. In contrast, Pix2seq quantizes each coordinate into one of the n_{bins} discrete bins, and represent each box with four tokens arranged sequentially. We adopt the similar idea and represent each box as four discrete tokens, $[<x_{\text{min}}>, <y_{\text{min}}>, <x_{\text{max}}>, <y_{\text{max}}>]$, where $<x>$,$<y>$ are quantized box tokens ranging from $<0>$, to $<n_{\text{bins}}-1>$.

Unified decoding sequence with <obj> token. We aim to have a unified decoding sequence s that can jointly represent text and box, meanwhile indicating word-box alignments. For the former, we unify the text and box vocabularies such that a single decoder can freely generate text or box tokens at any decoding step. Specifically, UniTAB's decoding vocabulary contains both text and box tokens, and has a size of $n_{\text{text}}+n_{\text{bins}}+2$. n_{text} and n_{bins} are the text vocabulary size and the number of coordinate bins. We use the same set of n_{bins} box tokens [10] for all four box coordinates. The output token selection at each decoding step is conducted over the entire unified vocabulary.

The remaining question is how to represent the word-box alignments in the output sequence. Instead of extra alignment score prediction [23,78,80,45], we represent word-box alignments inline with two introduced special tokens $<obj>$ and $<\backslash obj>$. Specifically, the model switches to box tokens right after any text words to be grounded, and inserts the $<obj>$ tokens before the first text word and after the last box tokens, respectively. For example, in Figure 2, we extend the text phrase "a donut" in the text-only caption as "$<obj>$ a donut <90> <83> <184> <180> $<\backslash obj>$" in the extended target sequence, where $90, 83, 184, 180$ are the quantized box coordinates for the blue box. The word-box alignments then can be easily extracted from the predicted sequence, *i.e.*, words and box within the pair of $<obj>$ tokens refer to the same entity, such as "a donut."

3.3 UniTAB Training

Objective. We train the model with a single language modeling objective [50], *i.e.*, at each decoding step t, maximizing the likelihood of target token s_t conditioned on input image v, input text l, and previous target tokens $s_{<t}$:

$$\mathcal{L}_{LM}(\theta) = -\sum_{t=1}^{T} \log P_\theta(s_t | s_{<t}, v, l), \tag{1}$$

where θ denotes the model parameters, and T is the target sequence length.

Training stages. UniTAB's unified structure facilitates the pre-training and finetuning that use the same language modeling objective. We train UniTAB with up to three stages. The first is vision-language pre-training, which leverages large-scale image-text dataset optionally with grounded box annotations. Then, we perform multi-task finetuning, where multiple downstream task datasets with supervised annotations are merged to finetune a single model for different VL tasks. Lastly, we could conduct task-specific finetuning that adapts the model to each specific task for further improvement. The three stages share the same training objective as in Eq. 1, but with different training corpus and input-output designs. We discuss the combinations of these different training stages in Section 4.3. We next introduce each of these three training stages.

1. Pre-training. Pre-training aims to use large-scale data loosely related to downstream tasks for general VL representation learning. We pre-train the model with a single language modeling objective to predict the target sequence s, conditioned on image v and input text l. We randomly set the input text l as an empty string or the text-only image description, with the same probability of 0.5. We train the model to generate the text+box sequence s shown in Figure 2. The model thus learns to perform both captioning-like (with empty string input) and grounding-like (with image description input) VL tasks during pre-training.

2. Multi-task finetuning. Multi-task finetuning [1,66,6] aims to use supervised annotations from multiple downstream task datasets to train a single model, thus avoiding task-specific model copies and further boosting the model performance. UniTAB's unified architecture and training objective facilitate the unique property of multi-task finetuning. Instead of having multiple duplicates of a pre-trained model, each optimized for a downstream task, multi-task finetuning trains a single set of parameters to perform all different VL tasks. We gather supervised data annotations from all 7 experimented VL tasks and train a single model with the language modeling objective. One major advantage of multi-task finetuning is that a single model can support multiple VL tasks, thus saving model parameters. Multi-task finetuning could also improve certain downstream tasks' performance by using annotations from different tasks.

3. Task-specific finetuning. UniTAB can also perform the standard task-specific finetuning as in VLP studies [42,12,38]. Furthermore, we observe that multi-task finetuning not only generates a single model that performs well in different VL tasks, but also serves as a good initialization point for a second-stage task-specific finetuning. We refer to this setting as "pre-finetuning" [1,66,6].

Inference. We use arg max sampling to obtain the sequence prediction. We then extract the text and box predictions from the sequence offline for final evaluation. For example, we discard box tokens to get the text prediction, and dequantize box tokens to get the box prediction. Finally, we evaluate the model on each downstream task with its desired output formats, *e.g.*, text for VQA, boxes for visual grounding, or both text and boxes for grounded captioning. We show in Section 4.3 that the task-agnostic output sequence design could help UniTAB generalize to unseen tasks that require text or box outputs.

4 Experiments

4.1 Experiment Overview

Downstream tasks. We evaluate UniTAB on 7 VL benchmarks (later summarized in Table 6). We start with grounded captioning [78,49] that requires the model to predict text, box, and their alignment. We then benchmark UniTAB on other representative VL tasks, including visual grounding [72,46,49], COCO image captioning [11], and VQAv2 visual question answering [5]. UniTAB approaches a wide range of VL tasks with a single unified architecture. In contrast, prior works require task-specific model designs, making it difficult to work on VL tasks with different desired output formats (text, box, or their combination).
Model variants. In addition to the comparison with state of the art, we systematically study the following UniTAB variants with different training stages:

- **Separate-scratch** conducts task-specific finetuning without pre-training.
- **Shared-scratch** conducts multi-task finetuning without pre-training.
- **Separate** is first pre-trained and then optimized separately for each downstream task, *i.e.*, the standard pretrain-then-finetune setting in VLP [42,12,38].
- **Shared** uses multi-task finetuning after pre-training, and shares a single set of parameters for all experimented VL tasks.
- **Pre-finetuning** adopts two-stage finetuning from a pre-trained checkpoint. The first stage is multi-task finetuning, followed by task-specific finetuning.

We take $\text{UniTAB}_{\text{Pre-finetuning}}$ as the default setting and refer to it as UniTAB. We report the main "Pre-finetuning" results in Section 4.2, and discuss the full results of UniTAB variants in Table 6 and Section 4.3.
Training corpus. The pre-training corpus [31] aggregates images from Flickr30k Entities [49], COCO [39,11], and Visual Genome (VG) [34] datasets. Text and grounded box annotations are from the referring expression datasets [72,46], VG regions, Flickr30k Entity annotations, and the GQA dataset [30]. The corpus contains around 200K images and 1.3M image-text pairs with grounded box annotations. Optionally, we further add the image-text data with no box annotations from Conceptual Captioning [55] and SBU [47] to pre-training, with settings and results detailed in Section 4.3. For multi-task finetuning, we collect supervised annotations from all 7 downstream datasets [78,49,72,46,11,5] to jointly train a single model for different tasks.
Implementation details. The transformer contains 6 encoder layers and 6 decoder layers, with 8 attention heads and a hidden dimension of 256 in each layer [8]. We use the scale and crop augmentation in DETR [8] such that the shortest side is between 480 and 800 pixels while the longest at most is 1333. We pre-train the model for 40 epochs, and finetune for 20 epochs in multi-task and task-specific settings. We use a learning rate of $1e^{-4}$ and $2e^{-5}$ for transformer layers and backbones. We train our model with AdamW [41] and adopt exponential moving average [61,31] with a decay rate of 0.9998 and a weight decay of $1e^{-4}$. More details are provided in Appendix A.

Table 2. Grounded image captioning results on the test set of Flickr30k Entities [49]. BLEU@4 [48], METEOR [18], CIDEr [62], and SPICE [3] metrics are used for caption evaluation. $F1_{all}$ and $F1_{loc}$ metrics [78] are used for grounding evaluation. Caption scores with † are optimized with CIDEr [54].

Method	Caption Eval.				Grounding Eval.	
	B@4	M	C	S	$F1_{all}$	$F1_{loc}$
NBT [44]	27.1	21.7	57.5	15.6	-	-
GVD [78]	27.3	22.5	62.3	16.5	7.55	22.2
Cyclical [45]	26.8	22.4	61.1	16.8	8.44	22.78
POS-SCAN [80]	$30.1^{\dagger}$	$22.6^{\dagger}$	$69.3^{\dagger}$	$16.8^{\dagger}$	7.17	17.49
Chen *et al.* [9]	27.2	22.5	62.5	16.5	7.91	21.54
UniTAB	**30.1**	**23.7**	**69.7**	**17.4**	**12.95**	**34.79**

4.2 Comparison with Prior Arts

Grounded captioning. The grounded captioning task [78,49] requires the model to generate a caption and ground all generated noun phrases to image regions. The final predictions consist of three parts, *i.e.*, the text caption, visual regions as boxes, and the grounding alignments between words and boxes. Instead of separately predicting those outputs with multiple output heads [78,45,80], UniTAB naturally represents all desired outputs with a single unified text+box output sequence. Following the established benchmarks [78,45,80] on the Flickr30k Entities dataset, we evaluate "captioning" and "grounding" separately with the caption metrics [48,18,3,62] and grounding F1 scores [78], respectively. The F1 score $F1_{all}$ evaluates grounding as a multi-label classification problem, where a correct prediction contains both the same object word as ground-truth (GT) caption and a larger than 0.5 IoU with the GT box. We also report $F1_{loc}$ that only computes the grounding score on correctly predicted object words.

Table 2 compares our method to state of the art [78,45,80,9]. We observe a significant improvement in the grounding quality, with the $F1_{all}$ score improving from 8.44 to 12.95, and $F1_{loc}$ from 22.78 to 34.79. UniTAB also achieves a better captioning quality, with the CIDEr score improving from 62.5 to 69.7, compared with prior arts [9]. By exploiting image-text data without box in pre-training, we further boost the CIDEr score from 69.7 to 74.2, as detailed in Section 4.3.

In addition to the performance improvement, UniTAB presents a simpler and more natural way for the grounded captioning task. Specifically, UniTAB does not require the pre-generated object regions [78,45,80] and avoids using multiple output heads. As shown in Figure 3(a), UniTAB naturally represents text, box, and word-region alignments in a single unified output sequence. Such a simple approach better transfers the model's grounding ability to other datasets or tasks with limited box or grounding annotations, such as COCO caption [11] and ImageNet [16], as shown in Figures 3(d,f). We hope UniTAB's new paradigm simplifies future studies on grounded VL tasks.

Visual grounding. Visual grounding aims to ground language queries into aligned image regions. We experiment on the sub-tasks of referring expression

Table 3. The performance comparisons (Acc@0.5) on the referring expression comprehension (Refcoco, Refcoco+, Refcocog) and phrase grounding task (Flickr30k Entities).

Method	Refcoco			Refcoco+			Refcocog		Flickr30k
	val	testA	testB	val	testA	testB	val-u	test-u	Entities
MAttNet [71]	76.40	80.43	69.28	64.93	70.26	56.00	66.67	67.01	-
FAOA [70]	72.05	74.81	67.59	55.72	60.37	48.54	59.03	58.70	68.71
TransVG [17]	81.02	82.72	78.35	64.82	70.70	56.94	68.67	67.73	79.10
ViLBERT [42]	-	-	-	72.34	78.53	62.61	-	-	-
UNITER [12]	81.41	87.04	74.17	75.90	81.45	66.70	74.02	68.67	-
VILLA [21]	82.39	87.48	74.84	76.17	81.54	66.84	76.18	76.71	-
MDETR [31]	86.75	89.58	81.41	79.52	84.09	70.62	81.64	80.89	**83.8**
$\text{UniTAB}_{\text{Separate}}$	86.32	88.84	80.61	78.70	83.22	69.48	79.96	79.97	79.39
UniTAB	**88.59**	**91.06**	**83.75**	**80.97**	**85.36**	**71.55**	**84.58**	**84.70**	79.58

comprehension [72,46] with Refcoco/Refcoco+/Refcocog, and phrase grounding [49] with Flickr30k Entities. Referring expression comprehension contains a query that describes a single image region and expects a single box prediction. Phrase grounding aims to ground all noun phrases in the input sentence, and requires the model to predict all referred boxes and the word-box alignments. In contrast to previous studies that do not know word-box alignments [71,70,17] or require separate alignment predictions [31], UniTAB generates a unified sequence with word-box alignments naturally represented by the special <*obj*> token. We report the standard metric Acc@0.5 [72,46,49].

As shown in Table 3, UniTAB outperforms the state of the art, including those pre-trained on larger VL corpus [42,12,21] and methods that use carefully-designed task-specific architectures [71,70,17]. Moreover, UniTAB's unified output with both text and box presents a more natural way of visual grounding, compared to box regression [70,17,31] or region index classification [71,12,13]. UniTAB's multi-task finetuning enables the use of data from different tasks and datasets, thus boosting performance on all splits, compared with $\text{UniTAB}_{\text{Separate}}$.

COCO captioning. We benchmark UniTAB on the COCO image captioning dataset [39]. We report the results without beam search [4] or CIDEr optimization [54]. Table 4 shows the captioning results on the Karparthy test split [32]. We refer to our pre-training corpus as "200K" in the "#Pre-train" column, and introduce the corpus used by compared methods later in Appendix A.

UniTAB achieves better performance than prior arts [68,13] that use similar amounts of pre-training images, with the CIDEr score improved from 117.3 to 119.8. Meanwhile, UniTAB does not require input region proposals or object tags [79,38,13]. Using extra image-text pairs [55,47] in pre-training further boosts the CIDEr score to 123.1. We expect a further gain by scaling up the pre-training corpus, as observed in VLP studies [74,36,65,27]. Despite only being evaluated with caption metrics on COCO, UniTAB's unified output sequence could also ground generated noun phrases to image regions, as visualized in Figure 3(d).

Visual question answering. UniTAB takes a generative approach to the VQA task [5], where the model generates a free-form text sequence to represent the

Table 4. COCO image captioning results on the Karparthy test split. The "#Pre-train" column shows the number of pre-training images, if any.

Method	#Pre-train	B@4	M	C	S
Unified VLP [79]	3M	36.5	28.4	117.7	21.3
OSCAR [38]	4M	36.5	30.3	123.7	23.1
E2E-VLP [68]	180K	36.2	-	117.3	-
VL-T5 [13]	180K	34.5	28.7	116.5	21.9
VL-BART [13]	180K	35.1	28.7	116.6	21.5
UniTAB	200K	36.1	28.6	119.8	21.7

Table 5. Visual question answering results on VQAv2 [5]. We experiment on both test-dev/test-std splits, and the Karpathy test split used in VL-T5 [13].

Method	#Pre-train	Test-		Karpathy-test		
		Dev	Std	In	Out	All
UNITER [12]	4M	72.7	72.9	74.4	10.0	70.5
VL-T5 [13]	180K	-	70.3	71.4	13.1	67.9
VL-BART [13]	180K	-	71.3	72.1	13.2	68.6
UniTAB	200K	70.7	71.0	71.1	11.1	67.5

Table 6. Summary of results obtained by UniTAB and its variants. The compared methods (upper portion) use task-specific architectures and training objectives, thus could only perform a subset of VL tasks. UniTAB (bottom portion) approaches all tasks with a unified framework and obtains competitive performance. The Refcoco/Refcoco+/Refcocog numbers are on the val set. The Flickr grounding and grounded caption results are on the test set. VQAv2-KP is the VQA Karpathy split [13]. $\text{UniTAB}_{\text{Pre-finetuning}}$ is the default setting that is also referred to as UniTAB.

Method	#Pre-train	Visual grounding				Grounded caption		COCO	VQAv2	
		Refcoco	Refcoco+	Refcocog	Flickr	Cider	$F1_{all}$	test-Cider	test-dev	KP-test
MDETR [31]	200K	86.75	79.52	81.64	83.8	-	-	-	70.6	-
UNITER [12]	4M	81.24	75.31	74.31	-	-	-	-	72.7	70.5
GVD [78]	-	-	-	-	-	62.3	7.55	-	-	-
VL-T5 [13]	180K	-	-	71.2	-	-	-	116.5	-	67.9
OSCAR [38]	4M	-	-	-	-	-	-	123.7	73.2	-
UniTAB Variants										
Separate-scratch	None	72.96	64.98	63.56	73.40	60.5	9.22	105.3	55.4	52.4
Shared-scratch	None	82.06	70.72	73.39	65.67	61.1	7.85	111.8	65.8	63.1
Separate	200K	86.32	78.70	79.96	79.39	65.6	11.46	119.3	69.9	66.6
Shared	200K	88.50	80.98	84.46	79.23	63.4	9.18	115.8	69.1	66.6
Pre-finetuning	200K	88.59	80.97	84.58	79.58	69.7	12.95	119.8	70.7	67.5

answer. Table 5 reports the VQA results on both the official test-dev/std split [5] and the Karparthy split [32] used in VL-T5 [13]. The Karparthy test set is further split into in- and out-domain subsets, based on whether the answer is covered in the top-K (K=3129) vocabulary [13]. The metric is the soft-voting accuracy [5]. UniTAB obtains competitive results to the state of the art, and performs better on the Karparthy out-of-domain subset than the discriminative approach [12].

4.3 Ablation and Analysis

Training stage ablation. We compare the variants of UniTAB to examine the influence of different pre-training and finetuning stages introduced in Section 3.3. The bottom portion of Table 6 summarizes the results. We first discuss the standard pretrain-then-finetune setting in VLP [42,12,38] that adopts task-specific finetuning. $\textbf{UniTAB}_{\textbf{Separate}}$ approaches various VL tasks with a single unified architecture, and obtains competitive results to the state of the art that has architectures tailored for each task, or uses larger-scale pre-training data. Compared with $\text{UniTAB}_{\text{Separate-scratch}}$ without pre-training, pre-training leads to consistent improvements on all experimented tasks.

With UniTAB's unified architecture and output modeling, we can train a single **UniTAB$_{\text{Shared}}$** model for all experimented VL tasks. Compared with UniTAB$_{\text{Separate}}$, the multi-task finetuning UniTAB$_{\text{Shared}}$ performs comparable or better on experimented VL tasks, while using **7** times fewer model parameters by avoiding task-specific model copies. The strong performance of UniTAB$_{\text{Shared}}$ indicates that we can use a single model for multiple downstream tasks, thus being *parameter efficient*. We further experiment with adding task-specific prefixes [66,13] to the input text. This variant uses a task-specific prefix such as "visual grounding:" to describe each sample's task. We observe that the task prefix has no major influence on model performance, as detailed in Appendix C.

In addition to achieving good performance with a single model, multi-task finetuning UniTAB$_{\text{Shared}}$ also provides a strong initialization point for further task-specific finetuning. **UniTAB$_{\text{Pre-finetuning}}$** further boosts the performance and achieves better or comparable performance than the state of the art on experimented VL tasks, as shown in the bottom row of Table 6.

Zero-shot generalization. The task-agnostic output sequence design helps UniTAB generalize to new tasks. UniTAB could perform certain tasks in a zero-shot manner by transferring the learned ability of generating text+box sequences s conditioned on image-text inputs. We next explore adapting UniTAB to ImageNet object localization [16]. Object localization [77,14,64] aims to localize an ImageNet class onto an object region. We take the words in class names as the text input, and have UniTAB generate text+box sequence s conditioned on image-text inputs. We then obtain box predictions by extracting boxes and alignments from s, similar to the phrase grounding post-processing. There exist two established benchmark settings. The "GT-known" [57,75,76,15] setting aims to localize a given ground-truth class. The metrics [14] "MaxBoxAcc" and "MaxBoxAccV2" are the Top-1 accuracy with an IoU threshold of 0.5, and the average at thresholds 0.3/0.5/0.7. The second setting tries to localize a predicted class. The metric is "Top-1 accuracy" with a 0.5 IoU threshold. We use EfficientNet [60] classification result with an accuracy of 77.5% for this setting.

We experiment with UniTAB$_{\text{Shared}}$ and show ImageNet object localization results in Table 7. UniTAB achieves better performance than the state of the art without using ImageNet images or annotations. The good generalization results show the possibility of generalizing UniTAB to unseen images and tasks in a zero-shot manner. We expect larger-scale pre-training to boost such generalization ability further, as observed in the NLP community [7,66].

Pre-training with additional image-text pairs. We experiment with adding image-text pairs without boxes in UniTAB pre-training, and examine if the extra image-text data could further improve VL tasks that require text output. For image-text pair data, we pre-train the model to generate the text-only caption conditioned on image and an empty text input. The model variant is referred to as "Separate††," which uses 4M image-text pairs from Conceptual Captioning [55] and SBU [47]. Table 8 compares "Separate††" with UniTAB$_{\text{Separate}}$ on grounded captioning, COCO captioning, and VQA. We observe consistent improvements in the text output quality by using extra image-text pairs, *i.e.*, +8.6 CIDEr score

Table 7. Zero-shot object localization results on ImageNet [16]. Prior works with the weakly supervised setting use ImageNet class labels.

Method	Top-1 Acc.	MaxBoxAcc	MaxBoxAccV2
CAM [77]	51.8	64.2	63.7
HaS [57]	49.9	63.1	63.4
CutMix [73]	51.5	65.4	63.3
MinMaxCAM [64]	-	66.7	65.7
UniTAB$_{\text{Shared}}$	**60.2**	**68.1**	**67.8**

Table 8. UniTAB pre-training with additional image-text pairs. "Separate††" uses additional 4M image-text pairs from CC [55] and SBU [47] that do not have grounded box annotations.

UniTAB	Grounded caption		COCO	VQAv2
	Cider	F1$_{all}$	test-Cider	KP-test
Separate	65.6	11.46	119.3	66.6
Separate††	74.2	12.62	123.1	69.1

on grounded captioning [49], +3.8 CIDEr score on COCO captioning [11], and +2.5% absolute accuracy on VQA [5]. Appendix C further discusses the benefit of pre-training with other addition data, such as boxes from object detection [39].

Model and output sequence design. We empirically observe that the introduced <*obj*> token not only naturally represents the word-box alignment, but also simplifies the sequence prediction by providing hints of the text-box code-switching, thus helping the VL tasks' performance. We postpone the detailed ablation studies on model and output sequence design to Appendix B, including the effectiveness of <*obj*> token, decoding sampling methods [4,25,10], the number of object tokens, decoding syntactic restrictions, *etc*.

4.4 Qualitative Results

Figure 3 shows the predictions made by UniTAB$_{\text{Shared}}$ on different VL tasks, where all predictions are made by a single model with the same set of parameters. On the right side of each subfigure, we show the input text and predicted output sequence. The output sequence is colored for visualization purposes only, where the text and box colors indicate the word-box alignments. We then show the extracted text and box predictions used for downstream task evaluation. For text, we discard all box tokens to obtain the text-only sequence. For boxes, we keep box tokens and dequantize them as box coordinate predictions [10].

UniTAB's task-agnostic output sequence seamlessly supports different VL tasks. Figure 3(a) shows an example of grounded captioning, where the input text is a blank string and both text and box predictions are used for evaluation. UniTAB could perform the phrase grounding task with the exact output sequence design, by replacing the blank input text with an image description, as shown in Figure 3(b). Figure 3(c) shows a referring expression comprehension example from the Refcocog dataset [46]. The model correctly localizes the referred "cat" in the "mirror." Despite not being used by the downstream task evaluation, the model successfully aligns the predicted box with phrase "the cat."

UniTAB's unified output sequence helps the model transfer the grounded description ability to datasets or tasks with limited box or grounding annotations. As shown in Figure 3(d), UniTAB learns grounded captioning on Flickr30k Entities and transfers such ability to COCO during multi-task finetuning. The generated caption not only has a good caption quality, as evaluated in Table 4,

Fig. 3. Predictions made by UniTAB$_{\text{Shared}}$ that uses a single model for different VL tasks. In each subfigure, we show the input text, the raw output sequence, and the extracted outputs for downstream task evaluations. Specifically, the output sequence contains an open-ended text sequence, box predictions (visualized as bounding boxes), and word-box alignments (visualized as the word-box colors). **(a-d)** UniTAB approaches a wide range of VL tasks with a single unified model and output sequence. **(e,f)** With the task-agnostic output sequence, we further generalize UniTAB to unseen images or even new tasks, with examples on ImageNet object localization and grounded captioning.

but also contains grounding predictions that make the description more comprehensive and interpretable. With the task-agnostic output sequence design, we further explore generalizing UniTAB to unseen images or even new tasks. Figure 3(e) shows an example of zero-shot object localization on ImageNet. The model correctly localizes the dog conditioned on the text input of ImageNet class label "brittany spaniel." Figure 3(f) shows an example of zero-shot grounded captioning on ImageNet images, where UniTAB generates a smooth caption and correctly grounds all noun phrases. More qualitative results are in Appendix D.

5 Conclusion

We have presented UniTAB that unifies text and box outputs for grounded VL modeling. With the special $<obj>$ token, UniTAB could generate both text and box predictions, with the word-box alignments naturally represented in the output sequence. Unifying text and box outputs allows the model to better approach grounded VL tasks such as grounded captioning. Furthermore, the unified multi-task network and the task-agnostic output sequence design make UniTAB parameter efficient and generalizable to new tasks. We see great potential in UniTAB, and believe it paves the way for building vision systems with stronger intelligence, such as in-context learning [7] and instruction tuning [66].

References

1. Aghajanyan, A., Gupta, A., Shrivastava, A., Chen, X., Zettlemoyer, L., Gupta, S.: Muppet: Massive multi-task representations with pre-finetuning. In: EMNLP (2021) 3, 7
2. Alberti, C., Ling, J., Collins, M., Reitter, D.: Fusion of detected objects in text for visual question answering. In: EMNLP (2019) 4
3. Anderson, P., Fernando, B., Johnson, M., Gould, S.: Spice: Semantic propositional image caption evaluation. In: ECCV (2016) 9
4. Anderson, P., He, X., Buehler, C., Teney, D., Johnson, M., Gould, S., Zhang, L.: Bottom-up and top-down attention for image captioning and visual question answering. In: CVPR (2018) 4, 10, 13
5. Antol, S., Agrawal, A., Lu, J., Mitchell, M., Batra, D., Zitnick, C.L., Parikh, D.: VQA: Visual Question Answering. In: ICCV (2015) 1, 3, 8, 10, 11, 13
6. Aribandi, V., Tay, Y., Schuster, T., Rao, J., Zheng, H.S., Mehta, S.V., Zhuang, H., Tran, V.Q., Bahri, D., Ni, J., et al.: Ext5: Towards extreme multi-task scaling for transfer learning. In: ICLR (2022) 3, 7
7. Brown, T.B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al.: Language models are few-shot learners. In: NeurIPS (2020) 12, 14
8. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: ECCV (2020) 3, 5, 6, 8
9. Chen, N., Pan, X., Chen, R., Yang, L., Lin, Z., Ren, Y., Yuan, H., Guo, X., Huang, F., Wang, W.: Distributed attention for grounded image captioning. In: ACMMM (2021) 3, 4, 9
10. Chen, T., Saxena, S., Li, L., Fleet, D.J., Hinton, G.: Pix2seq: A language modeling framework for object detection. In: ICLR (2022) 2, 3, 6, 13
11. Chen, X., Fang, H., Lin, T.Y., Vedantam, R., Gupta, S., Dollár, P., Zitnick, C.L.: Microsoft coco captions: Data collection and evaluation server. arXiv preprint arXiv:1504.00325 (2015) 1, 3, 8, 9, 13
12. Chen, Y.C., Li, L., Yu, L., Kholy, A.E., Ahmed, F., Gan, Z., Cheng, Y., Liu, J.: Uniter: Learning universal image-text representations. In: ECCV (2020) 4, 7, 8, 10, 11
13. Cho, J., Lei, J., Tan, H., Bansal, M.: Unifying vision-and-language tasks via text generation. In: ICML (2021) 2, 4, 5, 10, 11, 12
14. Choe, J., Oh, S.J., Lee, S., Chun, S., Akata, Z., Shim, H.: Evaluating weakly supervised object localization methods right. In: CVPR (2020) 12
15. Choe, J., Shim, H.: Attention-based dropout layer for weakly supervised object localization. In: CVPR (2019) 12
16. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: CVPR (2009) 1, 9, 12, 13
17. Deng, J., Yang, Z., Chen, T., Zhou, W., Li, H.: Transvg: End-to-end visual grounding with transformers. In: ICCV (2021) 10
18. Denkowski, M., Lavie, A.: Meteor universal: Language specific translation evaluation for any target language. In: Proceedings of the ninth workshop on statistical machine translation (2014) 9
19. Dou, Z.Y., Xu, Y., Gan, Z., Wang, J., Wang, S., Wang, L., Zhu, C., Liu, Z., Zeng, M., et al.: An empirical study of training end-to-end vision-and-language transformers. arXiv preprint arXiv:2111.02387 (2021) 4

20. Fu, J., Rui, Y.: Advances in deep learning approaches for image tagging. APSIPA Transactions on Signal and Information Processing **6** (2017) 1
21. Gan, Z., Chen, Y.C., Li, L., Zhu, C., Cheng, Y., Liu, J.: Large-scale adversarial training for vision-and-language representation learning. In: NeurIPS (2020) 10
22. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: CVPR (2014) 3
23. Gupta, T., Kamath, A., Kembhavi, A., Hoiem, D.: Towards general purpose vision systems. arXiv preprint arXiv:2104.00743 (2021) 2, 4, 5, 6
24. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016) 5
25. Holtzman, A., Buys, J., Du, L., Forbes, M., Choi, Y.: The curious case of neural text degeneration. In: ICLR (2020) 13
26. Hu, R., Singh, A.: Unit: Multimodal multitask learning with a unified transformer. In: ICCV (2021) 4, 5
27. Hu, X., Gan, Z., Wang, J., Yang, Z., Liu, Z., Lu, Y., Wang, L.: Scaling up vision-language pre-training for image captioning. In: CVPR (2022) 10
28. Huang, Z., Zeng, Z., Huang, Y., Liu, B., Fu, D., Fu, J.: Seeing out of the box: End-to-end pre-training for vision-language representation learning. In: CVPR (2021) 4
29. Huang, Z., Zeng, Z., Liu, B., Fu, D., Fu, J.: Pixel-bert: Aligning image pixels with text by deep multi-modal transformers. arXiv preprint arXiv:2004.00849 (2020) 4
30. Hudson, D.A., Manning, C.D.: Gqa: A new dataset for real-world visual reasoning and compositional question answering. In: CVPR (2019) 8
31. Kamath, A., Singh, M., LeCun, Y., Synnaeve, G., Misra, I., Carion, N.: Mdetr-modulated detection for end-to-end multi-modal understanding. In: ICCV (2021) 3, 4, 5, 8, 10, 11
32. Karpathy, A., Fei-Fei, L.: Deep visual-semantic alignments for generating image descriptions. In: CVPR (2015) 10, 11
33. Kim, W., Son, B., Kim, I.: Vilt: Vision-and-language transformer without convolution or region supervision. In: ICML (2021) 2, 4
34. Krishna, R., Zhu, Y., Groth, O., Johnson, J., Hata, K., Kravitz, J., Chen, S., Kalantidis, Y., Li, L.J., Shamma, D.A., et al.: Visual genome: Connecting language and vision using crowdsourced dense image annotations. IJCV (2017) 8
35. Li, G., Duan, N., Fang, Y., Gong, M., Jiang, D., Zhou, M.: Unicoder-vl: A universal encoder for vision and language by cross-modal pre-training. In: AAAI (2020) 4
36. Li, J., Selvaraju, R.R., Gotmare, A.D., Joty, S., Xiong, C., Hoi, S.: Align before fuse: Vision and language representation learning with momentum distillation. In: NeurIPS (2021) 4, 10
37. Li, L.H., Yatskar, M., Yin, D., Hsieh, C.J., Chang, K.W.: Visualbert: A simple and performant baseline for vision and language. arXiv preprint arXiv:1908.03557 (2019) 4
38. Li, X., Yin, X., Li, C., Hu, X., Zhang, P., Zhang, L., Wang, L., Hu, H., Dong, L., Wei, F., et al.: Oscar: Object-semantics aligned pre-training for vision-language tasks. In: ECCV (2020) 4, 7, 8, 10, 11
39. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: ECCV (2014) 1, 2, 8, 10, 13
40. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V.: Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692 (2019) 5

41. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101 (2017) 8
42. Lu, J., Batra, D., Parikh, D., Lee, S.: Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In: NeurIPS (2019) 4, 7, 8, 10, 11
43. Lu, J., Goswami, V., Rohrbach, M., Parikh, D., Lee, S.: 12-in-1: Multi-task vision and language representation learning. In: CVPR (2020) 4
44. Lu, J., Yang, J., Batra, D., Parikh, D.: Neural baby talk. In: CVPR (2018) 9
45. Ma, C.Y., Kalantidis, Y., AlRegib, G., Vajda, P., Rohrbach, M., Kira, Z.: Learning to generate grounded visual captions without localization supervision. In: ECCV (2020) 2, 3, 4, 6, 9
46. Mao, J., Huang, J., Toshev, A., Camburu, O., Yuille, A.L., Murphy, K.: Generation and comprehension of unambiguous object descriptions. In: CVPR (2016) 3, 8, 10, 13
47. Ordonez, V., Kulkarni, G., Berg, T.L.: Im2text: Describing images using 1 million captioned photographs. In: NeurIPS (2011) 8, 10, 12, 13
48. Papineni, K., Roukos, S., Ward, T., Zhu, W.J.: Bleu: a method for automatic evaluation of machine translation. In: ACL (2002) 9
49. Plummer, B.A., Wang, L., Cervantes, C.M., Caicedo, J.C., Hockenmaier, J., Lazebnik, S.: Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In: ICCV (2015) 2, 3, 4, 8, 9, 10, 13
50. Radford, A., Narasimhan, K., Salimans, T., Sutskever, I.: Improving language understanding by generative pre-training (2018) 3, 6
51. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J.: Exploring the limits of transfer learning with a unified text-to-text transformer. JMLR (2020) 6
52. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: CVPR (2016) 3, 6
53. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. In: NeurIPS (2015) 4, 5, 6
54. Rennie, S.J., Marcheret, E., Mroueh, Y., Ross, J., Goel, V.: Self-critical sequence training for image captioning. In: CVPR (2017) 9, 10
55. Sharma, P., Ding, N., Goodman, S., Soricut, R.: Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In: ACL (2018) 8, 10, 12, 13
56. Shen, S., Li, L.H., Tan, H., Bansal, M., Rohrbach, A., Chang, K.W., Yao, Z., Keutzer, K.: How much can clip benefit vision-and-language tasks? In: ICLR (2022) 4
57. Singh, K.K., Lee, Y.J.: Hide-and-seek: Forcing a network to be meticulous for weakly-supervised object and action localization. In: ICCV (2017) 12, 13
58. Su, W., Zhu, X., Cao, Y., Li, B., Lu, L., Wei, F., Dai, J.: Vl-bert: Pre-training of generic visual-linguistic representations. In: ICLR (2019) 4
59. Tan, H., Bansal, M.: Lxmert: Learning cross-modality encoder representations from transformers. In: EMNLP (2019) 4
60. Tan, M., Le, Q.: Efficientnet: Rethinking model scaling for convolutional neural networks. In: ICML (2019) 12
61. Tarvainen, A., Valpola, H.: Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In: NeurIPS (2017) 8
62. Vedantam, R., Lawrence Zitnick, C., Parikh, D.: Cider: Consensus-based image description evaluation. In: CVPR (2015) 9

63. Wang, J., Yang, Z., Hu, X., Li, L., Lin, K., Gan, Z., Liu, Z., Liu, C., Wang, L.: Git: A generative image-to-text transformer for vision and language. arXiv preprint arXiv:2205.14100 (2022) 4
64. Wang, K., Oramas, J., Tuytelaars, T.: Minmaxcam: Improving object coverage for cam-basedweakly supervised object localization. arXiv preprint arXiv:2104.14375 (2021) 12, 13
65. Wang, Z., Yu, J., Yu, A.W., Dai, Z., Tsvetkov, Y., Cao, Y.: Simvlm: Simple visual language model pretraining with weak supervision. In: ICLR (2022) 2, 4, 10
66. Wei, J., Bosma, M., Zhao, V.Y., Guu, K., Yu, A.W., Lester, B., Du, N., Dai, A.M., Le, Q.V.: Finetuned language models are zero-shot learners. In: ICLR (2022) 3, 7, 12, 14
67. Wikipedia contributors: Code-switching — Wikipedia, the free encyclopedia (2022), https://en.wikipedia.org/w/index.php?title=Code-switching&oldid=1068820985 3
68. Xu, H., Yan, M., Li, C., Bi, B., Huang, S., Xiao, W., Huang, F.: E2e-vlp: End-to-end vision-language pre-training enhanced by visual learning. In: ACL (2021) 10, 11
69. Xue, H., Huang, Y., Liu, B., Peng, H., Fu, J., Li, H., Luo, J.: Probing intermodality: Visual parsing with self-attention for vision-and-language pre-training. In: NeurIPS (2021) 4
70. Yang, Z., Gong, B., Wang, L., Huang, W., Yu, D., Luo, J.: A fast and accurate one-stage approach to visual grounding. In: ICCV (2019) 10
71. Yu, L., Lin, Z., Shen, X., Yang, J., Lu, X., Bansal, M., Berg, T.L.: Mattnet: Modular attention network for referring expression comprehension. In: CVPR (2018) 10
72. Yu, L., Poirson, P., Yang, S., Berg, A.C., Berg, T.L.: Modeling context in referring expressions. In: ECCV (2016) 1, 3, 8, 10
73. Yun, S., Han, D., Oh, S.J., Chun, S., Choe, J., Yoo, Y.: Cutmix: Regularization strategy to train strong classifiers with localizable features. In: CVPR (2019) 13
74. Zhang, P., Li, X., Hu, X., Yang, J., Zhang, L., Wang, L., Choi, Y., Gao, J.: Vinvl: Revisiting visual representations in vision-language models. In: CVPR (2021) 4, 10
75. Zhang, X., Wei, Y., Feng, J., Yang, Y., Huang, T.S.: Adversarial complementary learning for weakly supervised object localization. In: CVPR (2018) 12
76. Zhang, X., Wei, Y., Kang, G., Yang, Y., Huang, T.: Self-produced guidance for weakly-supervised object localization. In: ECCV (2018) 12
77. Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., Torralba, A.: Learning deep features for discriminative localization. In: CVPR (2016) 3, 12, 13
78. Zhou, L., Kalantidis, Y., Chen, X., Corso, J.J., Rohrbach, M.: Grounded video description. In: CVPR (2019) 2, 3, 4, 6, 8, 9, 11
79. Zhou, L., Palangi, H., Zhang, L., Hu, H., Corso, J.J., Gao, J.: Unified vision-language pre-training for image captioning and vqa. In: AAAI (2020) 4, 10, 11
80. Zhou, Y., Wang, M., Liu, D., Hu, Z., Zhang, H.: More grounded image captioning by distilling image-text matching model. In: CVPR (2020) 2, 3, 4, 6, 9