GLIPv2: Unifying Localization and VL Understanding

Haotian Zhang[*1†], **Pengchuan Zhang**[*2†♠], **Xiaowei Hu**[3], **Yen-Chun Chen**[3], **Liunian Harold Li**[4†]
Xiyang Dai[3], **Lijuan Wang**[3], **Lu Yuan**[3], **Jenq-Neng Hwang**[1], **Jianfeng Gao**[3]
[1]University of Washington, [2]Meta AI, [3]Microsoft, [4]UCLA
{haotiz,hwang}@uw.edu,pengchuanzhang@fb.com,liunian.harold.li@cs.ucla.edu,
{Xiaowei.Hu,Yen-Chun.Chen,Xiyang.Dai,lijuanw,luyuan,jfgao}@microsoft.com

Abstract

We present GLIPv2, a grounded VL understanding model, that serves both localization tasks (e.g., object detection, instance segmentation) and Vision-Language (VL) understanding tasks (e.g., VQA, image captioning). GLIPv2 elegantly unifies localization pre-training and Vision-Language Pre-training (VLP) with three pre-training tasks: phrase grounding as a VL reformulation of the detection task, region-word contrastive learning as a novel region-word level contrastive learning task, and the masked language modeling. This unification not only simplifies the previous multi-stage VLP procedure but also achieves mutual benefits between localization and understanding tasks. Experimental results show that a single GLIPv2 model (all model weights are shared) achieves near SoTA performance on various localization and understanding tasks. The model also shows (1) strong zero-shot and few-shot adaption performance on open-vocabulary object detection tasks and (2) superior grounding capability on VL understanding tasks. Code is released at `https://github.com/microsoft/GLIP`.

1 Introduction

Recently, a general interest arises in building general-purpose vision systems [24, 28, 66, 47], also called vision foundation models [6, 67], that solve various vision tasks simultaneously, such as image classification [35], object detection [44], and Visual-Language (VL) understanding [3, 11, 32]. Of particular interest, is the unification between *localization* tasks (e.g., object detection [44] and segmentation [8, 23]) and VL *understanding* tasks (e.g., VQA [3] and image captioning [11]). Localization pre-training benefits VL tasks [1, 70], and the "localization->VLP" two-stage pre-training procedure [46, 57, 13, 56, 39, 37, 75, 42, 40] is the common practice in VL community. A long-standing challenge is the unification of localization and understanding, which aims at *mutual* benefit between these two kinds of tasks, simplified pre-training procedure, and reduced pre-training cost.

However, these two kinds of tasks appear to be dramatically different: localization tasks are vision-only and require fine-grained output (e.g., bounding boxes or pixel masks), while VL understanding tasks emphasize fusion between two modalities and require high-level semantic outputs (e.g., answers or captions).

[24, 28, 66] have made early attempts at unifying these tasks in a straightforward multi-task manner, where a low-level visual encoder is shared across tasks, and two separate high-level branches are designed for localization and VL understanding, respectively. The localization tasks are still vision-only and do not benefit from the rich semantics in vision-language data. As a result, such unified models see the marginal mutual benefit or even performance degradation [28] compared with task-specific models.

[*]The two authors contributed equally. [†]Work done at Microsoft Research. ♠ Corresponding author.

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

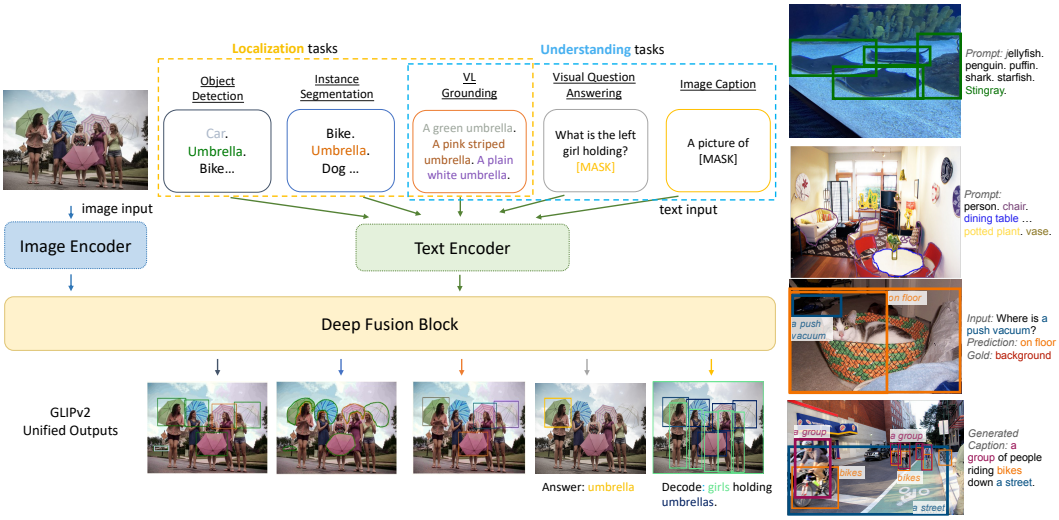

Figure 1: Left: GLIPv2, a pre-trained grounded VL understanding model, unifies various localization and VL understanding tasks. These two kinds of tasks mutually benefit each other, and enables new capabilities such as language-guided detection/segmentation and grounded VQA/captioning. Right: Additional examples from ODinW (detection), LVIS (segmentation), VQA, COCO Captioning.

In this paper, we identify "VL grounding" as a "meta"-capability for localization and understanding capabilities. VL grounding involves not only *understanding* an input sentence but also *localizing* the mentioned entities in the image (see an example in Figure 1). We build a **grounded VL understanding** model (GLIPv2) as a unified model for localization and VL understanding tasks.

Localization + VL understanding = grounded VL understanding. Localization tasks involve both localization and semantic classification, where classification can be cast as a VL understanding problem using the *classification-to-matching* trick (Section 3.1). Therefore, we reformulate localization tasks as VL grounding tasks, in which the language input is a synthesized sentence as the concatenation of category names [41]. Localization data are turned into VL grounding data, accordingly. The massive VL understanding data (image-text pairs) can be easily turned into VL grounding data in a self-training manner [41]. Therefore, GLIPv2 has a unified pre-training process: all task data are turned into grounding data and GLIPv2 is pre-trained to perform grounded VL understanding.

A stronger VL grounding task: inter-image region-word contrastive learning. GLIP [41] proposes the phrase grounding task as its pre-training task, which we argue is an easy task and does not fully utilize data information. For example, in the VL grounding task in Figure 1, the phrase grounding task only requires the model to match a given image region to one of the three phrases in the text input, i.e., "green, pink striped, or plain white umbrella?". This 1-in-3 choice is very easy, only requires color understanding, but loses lots of information in this grounding data: the umbrellas are not any other colors, like black, yellow, etc; objects in those regions are umbrellas but not any other categories, like car, bike, etc. From a contrastive learning view, this phrase grounding task only has two negatives. More negatives can be created from this annotation and thus enable stronger contrastive learning. In GLIPv2, we introduce the novel inter-image region-word contrastive learning task, which leverages phrases from other sentences in the same batch as potential negatives, as another much stronger VL grounding task. This new region-word contrastive loss enables GLIPv2 to learn more discriminative region-word features and demonstrates improvements over all downstream tasks.

GLIPv2 achieves mutual benefit between localization and VL understanding. 1) Experimental results (Table 2) show that a single GLIPv2 model (all model weights are shared) achieves near SoTA performance on various localization and understanding tasks. 2) Thanks to semantic-rich annotations from the image-text data, GLIPv2 shows superior zero-shot and few-shot transfer learning ability to open-world object detection and instance segmentation tasks, evaluated on the LVIS dataset and the "Object Detection in the Wild (ODinW)" benchmark. 3) GLIPv2 enables language-guided detection and segmentation ability, and achieves new SoTA performance on the Flick30K-entities phrase

grounding and PhraseCut referring image segmentation tasks. 4) Inherently a grounding model, GLIPv2 leads to VL understanding models with strong grounding ability, which are self-explainable and easy to debug. For example, GLIPv2, when GLIPv2 is finetuned on VQA, it can answer questions while localizing mentioned entities (see Figure 1 and Section 4.4).

2 Related Work

Localization models. Traditionally, localization tasks such as object detection and segmentation are single-modality and output bounding boxes or pixel masks [52, 43, 26, 14, 53, 10, 9]. One challenge of these single-modality models lies in generalization to rare and novel concepts: it is hard to collect localization data that cover many rare categories [23]. A long line of research focuses on this generalization problem, under the name of zero-shot [4, 76, 7, 77], weakly-supervised [19, 5, 61], or open-vocabulary [68, 22] localization. Built upon MDETR [30] and GLIP [41], GLIPv2 converts localization tasks into a grounded vision-language task using the classification-to-matching trick (Section 3). Thus GLIPv2 can learn from the semantic-rich vision-language data and shows strong performance on open-vocabulary localization tasks.

Vision-language understanding models. Vision-language (VL) understanding tasks such as VQA [3], image captioning [11], and image-text retrieval [31] involve understanding visual semantics and how they are expressed in natural language. Many VL models (e.g., BUTD) [2, 70] rely on a pre-trained localization model as their visual encoder; the downside is the pro-longed "localization->VLP" pre-training pipeline [46, 57, 13, 56, 39, 37, 75, 42, 40]. In contrast, GLIPv2 simplifies the pre-training pipeline and enables *grounded* VL understanding for better interpretability (Section 4.4).

Unifying localization and understanding. [24, 28, 66] made pioneering efforts in unifying localization and understanding. However, localization tasks are still treated as single-modality tasks, while VL tasks involve two modalities. The unification is achieved via straightforward multi-tasking: a low-level visual encoder is shared across tasks and two separate branches are designed for localization and VL understanding. Such unified models do not bring evident mutual benefit and often underperform task-specific models. In contrast, GLIPv2 identifies grounded VL understanding as a meta-task for localization and understanding. The task unification brings architecture unification: the unified grounded VL understanding model empowers a localization branch with VL capacity, arriving at a unified branch that excels at both tasks.

GLIPv2 vs GLIP. 1) GLIP shows that grounded pre-training improves localization. GLIPv2 further shows grounded pre-training improves VL understanding and thus leads to a unified model for localization and VL understanding. 2) GLIPv2 introduces the inter-image region-word contrastive loss, which is another and stronger grounding task than the pre-training task in GLIP. The proposed loss can be viewed as a region-word level generalization of the prevalent image-level contrastive learning [38, 51, 65]. 3) GLIPv2 outperforms GLIP on all benchmarks with the same pre-training data.

3 GLIPv2: Unifying Localization and VL Understanding

Based on the reformulation of object detection as a generalized phrase grounding task in GLIP [41], we unify both localization and VL understanding tasks as grounded vision-language tasks. A grounded vision-language task takes both image and text as inputs, and outputs region-level understanding results (e.g., detection, segmentation) and/or image-level understanding results with associated grounding/localization information (e.g., VQA, image captioning). We will present the unified grounded VL formulation and architecture in Section 3.1, the pre-training losses in Section 3.2, and transfer to downstream tasks in Section 3.3.

3.1 A Unified VL Formulation and Architecture

At the center of GLIPv2's unified formulation is the *classification-to-matching* trick, which reformulates any *task-specific fixed-vocab classification problem as an task-agnostic open-vocabulary vision-language matching* problem. The best example is the reformulation of image classification as image-text matching in CLIP [51], which enables the model to learn from raw image-text data

directly, and achieves strong zero-shot results on open-vocabulary classification tasks. In GLIPv2, we replace every semantic classification linear layer in traditional single-modality vision models with a vision-language matching dot-product layer.

As illustrated in Figure 1, GLIPv2's unified VL architecture is based on the generic architecture we term Architecture **Π**. It consists of a dual encoder, denoted as Enc_V and Enc_L, and a fusion encoder, denoted as Enc_{VL}. The model takes an image-text pair $(\text{Img}, \text{Text})$ as input, and extract visual and text features as below:

$$\mathring{O} = \text{Enc}_V(\text{Img}), \quad \mathring{P} = \text{Enc}_L(\text{Text}), \quad O, P = \text{Enc}_{VL}(\mathring{O}, \mathring{P}), \tag{1}$$

where $(\mathring{O}, \mathring{P})$ and (O, P) denote the image/text features *before* and *after* VL fusion, respectively.

Vision-Language understanding tasks. Arch **Π** is the most popular model architecture for VL understanding tasks. Given the cross-modality fused representations O and P, it is straightforward to add lightweight task-specific heads for various VL tasks. For example, GLIPv2 adds a two-layer MLP on top of text features P as the masked language modeling (MLM) head, to perform the MLM pre-training. We provide model details of VQA and image captioning in Section 3.3.

(Language-guided) object detection and phrase grounding. Following GLIP [41], GLIPv2 uses the classification-to-matching trick to unify detection and grounding. More specifically, for detection, we simply replace the class logits $S_{\text{cls}} = OW^T$, where W is the weight matrix of the box classifier, with a task-agnostic region-word similarity logits $S_{\text{ground}} = OP^T$, where text features P are label embeddings from a task-agnostic language encoder. As shown in Figure 1, object detection and phrase grounding share the same input/output format and model architecture. See GLIP [41] for more details. Their only difference is the input text format: (1) for object detection, the text input is a string of concatenated candidate object labels; (2) for phrase grounding, the text input is a natural language sentence. We refer to GLIP [41] for more details.

(Language-guided) instance segmentation and referring image segmentation. Given the object detection results, an instance segmentation head is added to classify each pixel within the box into a semantic class. Again, GLIPv2 uses the classification-to-matching trick to produce a unified instance segmentation head for the standard instance segmentation tasks and the referring image segmentation tasks and leverage both types of data for its pre-training. This classification-to-matching trick can also apply to many other semantic classification heads in single modality CV models (e.g., semantic segmentation) and thus transfers them to language-guided CV models.

3.2 GLIPv2 Pre-training

The GLIPv2 is pre-trained with three pre-training losses: phrase grounding loss $\mathcal{L}_{\text{ground}}$ from a vision-language reformulation of the object detection task, region-word contrastive loss $\mathcal{L}_{\text{inter}}$ from a novel region-word level contrastive learning task, and the standard masked language modeling loss $\mathcal{L}_{\text{mlm}}$ proposed in BERT [17].

$$\mathcal{L}_{\text{GLIPv2}} = \underbrace{\mathcal{L}_{\text{loc}} + \mathcal{L}_{\text{intra}}}_{\mathcal{L}_{\text{ground}}} + \mathcal{L}_{\text{inter}} + \mathcal{L}_{\text{mlm}} \tag{2}$$

Similar to losses in detection tasks, the grounding loss $\mathcal{L}_{\text{ground}}$ has two parts: the localization loss $\mathcal{L}_{\text{loc}}$ trains localization heads with bounding-box supervision, e.g., RPN loss, box regression loss and/or centerness loss [59]; the intra-image region-word alignment loss $\mathcal{L}_{\text{intra}}$ is essentially the semantic classification/retrieval loss for each region.

Intra-image region-word alignment loss. Given one image-text pair $(\text{Img}, \text{Text})$, we obtain the image and text features *after* cross-modality fusion O and P. The Intra-image region-word alignment loss is computed by

$$\mathcal{L}_{\text{intra}} = loss(OP^T; T), \tag{3}$$

where OP^T is the similarity score between image regions and word tokens, and T is the target affinity matrix determined by the ground-truth annotations. The loss function $loss$ is typically a cross-entropy loss for two-stage detectors [53] and a focal loss [43] for one-stage detectors.

However, as discussed in Section 1, this intra-image region-word contrastive learning is rather weak in the sense of contrastive learning, due to the limited number of phrases that can one caption can

contain. GLIP [41] alleviates this problem by appending a few negative sentences to form a longer text input with more (negative) phrases. However, constrained by the maximal length of text tokens (256 in GLIP and GLIPv2), only a few negative sentences can be added and the number of negative phrases remains in the order of 10's. This small-negative-example problem also exists in detection data [41] when the input text cannot include all class names in a detection dataset, e.g., Objects365.

Inter-image region-word contrastive loss. In GLIPv2, we propose using phrases from other image-text pairs in the same batch as negative examples, which effectively increases the number of negative examples to the order of 1000's, with nearly negligible additional computational cost.

As in (1), given a batch of image-text pairs $(\text{Img}^i, \text{Text}^i)_{i=1}^B$ and their ground-truth annotations $(T^i)_{i=1}^B$, the model produces the image and text features *before* and *after* VL fusion, denoted as $(\mathring{O}^i, \mathring{P}^i)_{i=1}^B$ and $(O^i, P^i)_{i=1}^B$, respectively. Then as illustrated in Figure 2 (Left), a batch-wise similarity matrix $S_{\text{ground}}^{\text{batch}}$ and a batch-wise target affinity matrix T^{batch} are constructed by considering all the image regions and text phrases across this batch. Their (i, j)'th blocks are obtained as below:

$$S_{\text{ground}}^{\text{batch}}[i,j] = \mathring{O}^i(\mathring{P}^j)^T, \quad T^{\text{batch}}[i,j] = \begin{cases} T^i, & \text{if } i = j \\ \text{obtained by label propagation}, & \text{otherwise.} \end{cases} \tag{4}$$

The inter-image region-word contrastive loss is then defined as the standard bi-directional contrastive loss applied on all image regions and phrases in this batch:

$$\mathcal{L}_{\text{inter}} = \text{cross_entropy_loss}(S_{\text{ground}}^{\text{batch}}, T^{\text{batch}}, \text{axis} = 0) + \text{cross_entropy_loss}(S_{\text{ground}}^{\text{batch}}, T^{\text{batch}}, \text{axis} = 1). \tag{5}$$

Compared with that in the inter-image contrastive loss (3), the number of negatives is multiplied by batch size B in this inter-image contrastive loss (5). We elaborate two important details in (4). (1) GLIPv2 uses the image text features $(\mathring{O}^i, \mathring{P}^i)_{i=1}^B$ before VL fusion, *not* $(O^i, P^i)_{i=1}^B$ after VL fusion, to compute the batch-wise similarity matrix in the inter-image contrastive loss (4). Otherwise, the image and text features after VL fusion would have seen the paired information (1), and thus the model can easily rule out the negatives from misaligned images/texts. (2) We cannot simply assign all regions and texts from unpaired image-text as negative pairs, as done in the standard contrastive loss in CLIP [51]. Instead, we determine the off-diagonal blocks in the target affinity matrix T^{batch} by *label propagation*. For example, as illustrated in Figure 2 (Left), if a region is annotated as "person", it should be a positive pair with all "person" phrases in detection-type texts. We do not propagate positives to grounding-type texts (natural sentences) because phrases in sentences carry contexts that are unique to that image-sentence pair.

Pre-training with both detection and paired-image-text data. GLIPv2 pre-training data is in the image-text-target triplet format $(\text{Img}, \text{Text}, T)$, where the target affinity matrix T contains the box-label localization annotations. We also use massive image-text pair data $(\text{Img}, \text{Text})$ to pre-train GLIPv2, by generating grounding boxes $\hat{T}$ for phrases in the text with the GLIP pre-trained model from [41]. The human-annotated OD/grounding data provides high-fidelity localization supervision, while the massive image-text data greatly improves the concept diversity for GLIPv2.

Second-stage pre-training of the segmentation head. GLIPv2 performs a second-stage pre-training of the language-guided segmentation head on both instance segmentation and image referring segmentation data, while fixing all other parts of the model.

3.3 Transfer GLIPv2 to Localization and VL Tasks

We introduce two ways to easily transfer GLIPv2 to various downstream tasks. In addition, GLIPv2 can perform conventional VL tasks (e.g., VQA) along with localization, effectively making every task we consider a "grounded VL understanding" task.

One model architecture for all. GLIPv2 can be transferred to downstream tasks by fine-tuning the model with an (optional) task-specific head. 1) For *detection and segmentation* tasks, no task-specific head is needed as the pre-training architecture can inherently perform detection and segmentation. 2) For *VL* tasks: for VQA, a classification head is added on top of the hidden representation of the start-of-sequence token; for caption generation, we train with a unidirectional language modeling loss, which maximizes the likelihood of the next word given context. We use a unidirectional attention mask and prevent the image part from attending to the text in the fusion layers.

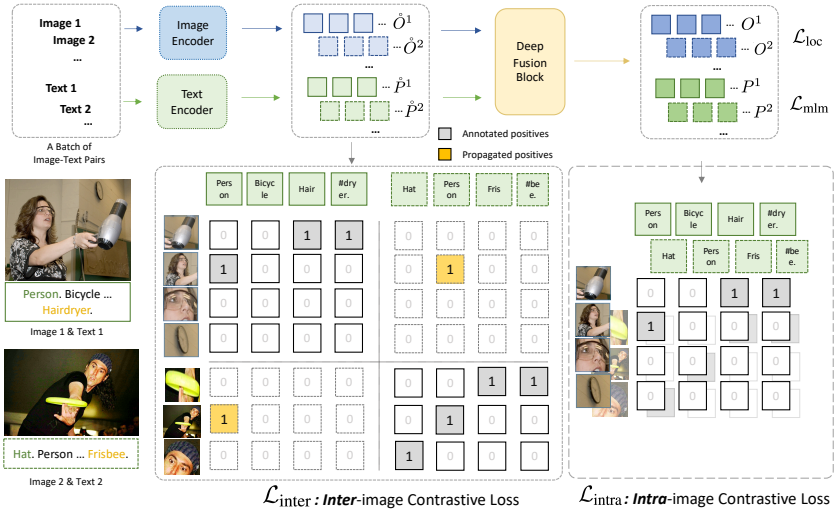

Figure 2: GLIPv2 pre-training losses: the intra-image alignment loss $\mathcal{L}_{\text{intra}}$ (right) takes features after VL fusion and compute loss over region-word pairs within each image-text pair; the inter-image contrastive loss (left) $\mathcal{L}_{\text{inter}}$ takes features before VL fusion and computes loss over all region-word pairs across a batch of image-text pairs. Label propagation is used to determine the off-diagonal blocks of the $\mathcal{L}_{\text{inter}}$ target matrix (4).

One set of weights for all. There is a growing interest in developing models that can be transferred to various tasks while only changing the least amount of parameters to save training time and storage cost [55, 36]. Following GLIP, GLIPv2 can be transferred to localization tasks in a *zero-shot* or a *prompt-tuning* setting (Section 4.2). One single GLIPv2 model can serve various tasks, where each task only keeps few or no parameters. Of particular interest is the prompt tuning setting. For a certain localization task, the text prompt is the same for all input images; thus, we could directly tune $\mathring{P}$, a small prompt embedding matrix, to adapt GLIPv2 to new tasks. Prompt tuning in a deep-fused model such as GLIPv2 is different from the conventional linear probing/prompt tuning setting [62, 51, 73] in shallow-interacting vision models such as CLIP. The latter can also be viewed as only tuning a small prompt/softmax embedding P; however, tuning P only affects the very last layer of the model while the visual representation is still frozen. In contrast, GLIP/GLIPv2's visual representation is conditioned on the prompt embedding $\mathring{P}$; tuning $\mathring{P}$ changes the text, visual, as well as fused embeddings. As a result, prompt tuning in GLIPv2 is highly effective, often matching the performance of fine-tuning (see Table 2). This is in contrast to the common observation in CV that linear probing lags behind fine-tuning by a large gap [25].

Grounded VL understanding. GLIPv2 also enables grounded VL understanding, where we retain the ability to perform grounding when fine-tuning the model to a downstream VL task. This increases the interpretability of the model. Specifically, we first turn the VL data of the downstream task into grounded VL data using a pre-trained GLIP model. Then we train the model with both the downstream task head and grounding head. For VQA, the model is trained to predict the answer and ground entities in the question as well as the implied entity in the answer; for captioning, the model is trained to predict the next word given the context and ground the current decoded word. By tuning localization tasks into a grounded VL task and augmenting VL tasks with grounding ability, we effectively turn every task into a grounded VL understanding task (see examples in Figure 1).

4 Experiments

In this section, we show that GLIPv2 serves as a performant and easy-to-deploy general-purpose vision system. 1) **One Model Architecture for All** (Section 4.1). GLIPv2 can be directly fine-tuned to both localization and VL understanding tasks with minimal architecture change. It achieves performance on par with SOTA models with specialized architectures. 2) **One Model Weight for All** (Section 4.2). GLIPv2 can be transferred to localization tasks in a zero-shot manner with zero

Model	Model Type	COCO-Det (test-dev)	ODinW (test)	LVIS (minival)	COCO-Mask (test-dev)	Flickr30K (test)	PhraseCut (test)	VQA (test-dev / test-std)	Captioning (Karpathy-test)
Mask R-CNN [26]	Localization	39.8	-	33.3 / -	- / 37.1	-	-	-	-
DETR [9]		42.0	-	17.8 / -	-	-	-	-	-
DyHead-T [15]		49.7	60.8	-	-	-	-	-	-
DyHead-L [15]		60.3*	-	-	-	-	-	-	-
VisualBERT [39]	Understanding	-	-	-	-	71.33	-	70.8 / 71.0	-
UNITER [12]		-	-	-	-	-	-	73.8 / 74.0	-
VinVL [70]		-	-	-	-	-	-	**76.5 / 76.6**	130.8
GPV [24]	Localization & Understanding	-	-	-	-	-	-	62.5 / -	102.3
UniT [28]		42.3	-	-	-	-	-	67.6 / -	-
MDETR [30]		-	-	24.2 / -	-	84.3	53.7	70.6 / 70.6	-
Unicorn [66]		-	-	-	-	80.4	-	69.2 / 69.4	119.1
GLIP-T [41]	Localization & Understanding	55.2	64.9	-	-	85.7	-	-	-
GLIP-L [41]		61.5*	68.9	-	-	87.1	-	-	-
GLIPv2-T (**Ours**)	Localization & Understanding	55.5	66.5	50.6 / 41.4	53.5 / 42.0	86.5	59.4	71.6 / 71.8	122.1
GLIPv2-B (**Ours**)		58.8	69.4	57.3 / 46.2	59.0 / 45.8	87.5	**61.3**	73.1 / 73.3	128.5
GLIPv2-H (**Ours**)		**60.6 (62.4*)**	**70.4**	**59.8 / 48.8**	**59.8 / 48.9**	**87.7**	**61.3**	74.6 / 74.8	**131.0**

Table 1: One model architecture results. For COCO-Det test-dev, * indicates multi-scale evaluation. For LVIS, we report the numbers for both `bbox` and `segm` on minival to avoid data contamination due to the pre-training. For Flickr30K test, we report the metric under `R@1`. For COCO-Mask, we also report both `bbox` and `segm` on test-dev.

parameter update; with prompt tuning, a single GLIPv2 model can achieve comparable performance with fully fine-tuned settings on both localization and understanding tasks.

Following GLIP [41], we adopt Swin Transformer [45] as the image encoder Enc_V, text transformers [60, 51] as the text encoder Enc_L, Dynamic Head [15] with language-aware deep fusion [41] as the fusion encoder Enc_{VL}, and Hourglass network [49] as instance segmentation head feature extractor. We train GLIPv2 at three scales: GLIPv2-T, GLIPv2-B, and GLIPv2-H.

GLIPv2-T has the same model config and initialization as GLIP-T: Swin-Tiny and BERT-Base as the dual encoder. The model is pre-trained on the following data: 1) O365, 2) GoldG as in GLIP-T (C), and 3) Cap4M, 4M image-text pairs collected from the web with boxes generated by GLIP-T [41]. **GLIPv2-B/GLIPv2-H** are based on Swin-Base/Swin-Huge and the pre-layernorm text transformer [18] as dual encoder, and are initialized from the UniCL [65] checkpoints. We observe much stabler training with GPT-type pre-layernorm transformer [18] than BERT-type post-layernorm transformer. The training data contain: 1) FiveODs (2.78M data) [1]; 2) GoldG as in MDETR [30]; and 3) CC15M+SBU, 16M public image-text data with generated boxes by GLIP-L [41]. **Segmentation heads** of GLIPv2 models are pre-trained on COCO, LVIS [23] and PhraseCut [63], with all other model parameters are frozen.

Note All datasets above were collected by the creators (cited) and consent for any personally identifiable information (PII) was ascertained by the authors where necessary. Due to limited space, we refer to supplementary for details of training recipes and hyper-parameters.

4.1 One Model Architecture for All

We compare GLIPv2 to existing object detection and vision-language pre-training methods on a wide range of tasks. We fine-tune the model on 8 different downstream tasks and report the performance in Table 1. We make the following observations.

GLIPv2 v.s. specialized Localization methods. GLIPv2 outperforms previous localization models on generalization to both common and rare classes and domains *with a single model architecture and pre-training stage*. *1) OD on common categories (COCO-Det)*, GLIPv2-T achieves 5.8 improvement compared to the standard DyHead-T trained on O365 (55.5 v.s. 49.7). GLIPv2-H reaches 62.4 AP on test-dev, and surpass the performance of the previous SoTA model GLIP-L. *2) OD on rare / unseen categories (LVIS)*, GLIPv2-T outperforms a supervised MDETR on the `bbox` by a great margin (59.8 v.s. 24.2). *3) Generalization to diverse real-word tasks (ODinw)*, GLIPv2-T (55.5) performs better than original GLIP-T (64.9) on the average of 13 public datasets; GLIPv2-B outperforms GLIP-L by 0.5 AP. *4) Instance segmentation (COCO-Mask & PhraseCut)*, for traditional instance segmentation

[1] Besides O365, it combines with 4 additional OD datasets including COCO [44], OpenImages [33], Visual Genome [34], and ImageNetBoxes [35]

Model	Direct Evaluation				Prompt Tuning				
	COCO-Mask (minival)	ODinW (test)	LVIS-Det (minival)	Flickr30K (minival)	COCO-Det (test-dev)	ODinW (test)	LVIS (minival)	COCO-Mask (test-dev)	PhraseCut (test)
GLIP-T	46.6/–	46.5	26.0	85.7	–	46.5	-	-	-
GLIP-L	49.8/–	52.1	37.3	87.1	58.8	67.9	-	-	-
GLIPv2-T	**47.3**/35.7	48.5	**29.0**	86.0	53.4 (-2.1)	64.8 (-1.7)	49.3 / 34.8 (-1.3 / -6.6)	53.2 / 41.2 (-0.3 / -0.8)	49.4
GLIPv2-B	61.9†/43.4	54.2	48.5	87.2	59.0 (+0.2)	67.3 (-2.1)	56.8 / 41.7 (-0.5 / -4.5)	58.8 / 44.9 (-0.2 / -0.9)	55.9
GLIPv2-H	64.1†/47.4	**55.5**	50.1	**87.7**	**60.2 / 61.9*** (-0.4 / -0.5)	**69.1** (-1.3)	**59.2 / 43.2** (-0.6 / -5.7)	**59.8 / 47.2** (-0.0 / -1.7)	**56.1**

Table 2: One set of weights results v.s. Original GLIP. * indicates multi-scale evaluation. Numbers in red clearly points out the difference between the prompt tuning and full fine-tuning results (see Table 1). Numbers in gray mean that they are not in *zero-shot* manner. †: these two numbers are artificially high due to some overlap between COCO-minival and VisualGenome-train.

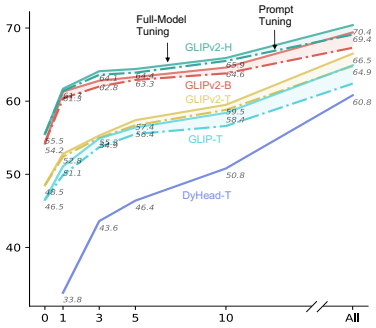

Figure 3: Data efficiency of GLIPv2 on ODinW. The X-axis is the amount of task-specific data, from zero-shot to all data. Y-axis is the average AP across 13 datasets.

Model	Zero-Shot	Prompt Tuning / Fine Tuning				
	0	1	3	5	10	All
DyHead-T O365 [41]	-	- / 33.8	- / 43.6	- / 46.4	- / 50.8	- / 60.8
$\mathcal{L}_{\text{loc}} + \mathcal{L}_{\text{intra}}$ (GLIP-T)	46.5	49.9 / 51.3	53.7 / 54.9	55.5 / 56.4	56.6 / 58.4	62.4 / 64.9
$\mathcal{L}_{\text{loc}} + \mathcal{L}_{\text{intra}} + \mathcal{L}_{\text{inter}}$	48.4	52.1 / 51.4	55.6 / 55.3	56.7 / 56.6	58.3 / 59.5	62.9 / 66.3
$\mathcal{L}_{\text{loc}} + \mathcal{L}_{\text{intra}} + \mathcal{L}_{\text{inter}} + \mathcal{L}_{\text{mlm}}$	48.5	52.4 / 52.8	55.6 / 55.6	57.4 / 57.4	58.8 / 59.7	64.8 / 66.5

Table 3: Zero-shot, prompt tuning, and full fine-tuning performance on ODinW. GLIPv2 models exhibit superior data efficiency.

(i.e., COCO-Mask), GLIPv2-H outperforms the well-known Mask R-CNN by a great margin on `segm`. For language-guided segmentation (i.e., PhraseCut), compared to MDETR, GLIPv2-T achieves an improvement of 5.7 mask AP.

GLIPv2 v.s. specialized VL Understanding methods. GLIPv2 rivals with SoTA specialized models for VL tasks. *1) For VQA*, GLIPv2 outperforms VisualBERT and UNITER and approaches the previous SoTA model VinVL. *2) For Captioning*, the best GLIPv2 even surpasses VinVL (VinVL and GLIPv2 are not trained with CIDEr optimization).

GLIPv2 v.s. localization and VL models. Prior works such GPV, UniT and Unicorn have also explored unifying localization and VL models (see a discussion in Section 2). GLIPv2 outperforms all previous systems on both localization and VL tasks. For the best GLIPv2-H, it outperforms the UniT by a great margin (18.3 AP) on COCO object detection tasks. Meanwhile, it also surpasses UniT's performance on VQA by 6.9 points and GPV's peformance on Image Captioning as well.

Takeaway. Most notably, GLIPv2 outperforms previous "unified" models (GPV, UniT, MDETR, Unicorn) by a large margin. This is the first time that a single model architecture could achieve near SoTA performance on both localization and understanding. In contrast, in prior work, there exists certain trade-off between localization and understanding: models that aim to achieve high understanding performance tend to have lower localization performance (e.g., UNiT's detection performance is limited to the DETR [9] architecture), as it is not trivial to merge a SoTA localization branch and a SoTA VL branch into a single model.

4.2 One Set of Model Parameters for All

GLIPv2 is pre-trained to perform grounding; thus it can be transferred to various localization tasks with changing zero or few parameters. We evaluate GLIPv2 under two such settings: 1) direct evaluation, where we transfer the model "as is" without any parameter change, and 2) prompt tuning, where only the prompt embedding is tuned for specific tasks (Section 3.3).

Direct evaluation. The pre-trained GLIPv2 can be directly evaluated on any object detection task (by concatenating the object categories into a text prompt) and visual grounding task without any further tuning. We evaluate the models on four localization tasks: COCO, ODinW, LVIS, and Flickr30, and their results are presented in Table 2. Note that for GLIPv2-B and GLIPv2-H, the training sets of Flick30K and LVIS are present in the pre-training data. Thus, reported numbers on these metrics are not *zero-shot* evaluation (we have marked them gray). For all other evaluation results, the models are evaluated in *zero-shot* settings without any further tuning.

GLIPv2 can be effortlessly transferred to different localization tasks without further tuning. 1) For *COCO*, GLIPv2-T achieves a zero-shot performance of 47.3 without seeing any COCO training images. This surpasses well-established supervised systems (e.g., Mask R-CNN) and also outperforms GLIP-T by 0.7 AP. 2) For *ODinW*, GLIPv2 also shows strong zero-shot performance. GLIPv2-T (48.5) surpasses the GLIP-T (46.5). Meanwhile, the zero-shot performance of GLIPv2-B and GLIPv2-H even surpasses the 10-shot tuning performance of DyHead-T (to be introduced in Figure 3). 3) For *LVIS*, GLIPv2-T achieves a 3 AP improvement performance compared to the GLIP-T. 4) For *Flickr30K*, GLIPv2-B achieves even higher number (87.2) compared to original GLIP-L (87.1).

Prompt Tuning. Following GLIP, GLIPv2 supports efficient prompt tuning: the visual representation is heavily conditioned on the text representation due to the deep fusion block (Section 3.3); thus we could fine-tune only the prompt embedding for each task but still maintain high performance.

Prompt tuning GLIPv2 achieves similar performance as full fine-tuning. When comparing the performance of each task in Table 1 and 2 at the same time, for GLIPv2, prompt tuning performance almost matches the one model architecture results on localization tasks, without changing any of the grounding model parameters.

4.3 GLIPv2 as a Strong Few-Shot Learner

We demonstrate GLIPv2's performance on ODinW datasets with respect to different amounts of training data in Figure 3. The performance improvement between GLIPv2-T and GLIP-T exhibits more superior data efficiency for prompt tuning. We compare with the SoTA detector DyHead-T, pre-trained on Objects365 in Table 3. It can be seen that a zero-shot GLIPv2-T (48.5) outperforms a outperforms 5-shot DyHead-T (46.4) while the performance of one-shot GLIPv2-H (61.3) surpasses a all-shot fully supervised DyHead-T (60.8).

4.4 Analysis

Pre-training losses Table 4 shows the performance of the downstream tasks with different variants of our method. Compared to the GLIP pre-training tasks with only intra-image region-word contrastive loss (Row 3), adding inter-image word-region loss (Row 5) substantially improves the pre-trained model performance across all the object detection tasks (COCO, ODinW, and LVIS) on both zero-shot and fine-tuned manner. Consistent with common observations from most VL understanding methods, adding MLM loss (Row4) benefits for learning the representation for understanding tasks (Flick30k, VQA, and Captioning). Furthermore, using all three losses together at the 1st stage pre-training and doing the 2nd stage pre-training without MLM on OD and GoldG data, GLIPv2 (Row6) can perform well on both the localization and VL understanding tasks.

An additional stage of pre-training is applied for small models (GLIPv2-T and GLIPv2-B) due to limited model capacity. In order to achieve higher performance on both localization and understanding tasks, we find that including all data (even with some noise) and MLM loss in the first stage of pre-training will benefit the model for learning a better representation of both localization and understanding capability. Since the OD tasks require the model with more accurate localization ability, in our 2nd stage of pre-training, we decide to eliminate the MLM loss. The large model (GLIPv2-H) does not need this additional stage because it has enough capacity to learn both word-region alignment and MLM together in a single stage.

Pre-training data Table 5 reports the last checkpoint results on GLIPv2 when we do the scaling up of pre-training data. As more weak image-text pair data (Cap) is involved in our training, it benefits both standard/in-domain (i.e., COCO, Flickr30K) and large-domain gap (i.e., ODinW, LVIS) tasks. We also show that by adding the inter-image region-word contrastive helps when we are fixing the data at the same scale. For large-domain gap tasks, adding the inter-image region-word contrastive

Row	Model	COCO	ODinW	LVIS	Flickr30K	VQA	Captioning
1	No pre-train	–/50.6	–/60.8	–	–	64.6	111.5
2	+ $\mathcal{L}_{\text{mlm}}$	–/48.5	–/37.4	–	–	64.6	110.9
3	+ $\mathcal{L}_{\text{loc}} + \mathcal{L}_{\text{intra}}$	46.6/55.2	46.5/64.9	26.0	85.7	69.4	119.7
4	+ $\mathcal{L}_{\text{loc}} + \mathcal{L}_{\text{intra}} + \mathcal{L}_{\text{mlm}}$	47.0/55.2	47.6/66.2	28.5	86.5	69.8	120.7
5	+ $\mathcal{L}_{\text{loc}} + \mathcal{L}_{\text{intra}} + \mathcal{L}_{\text{inter}}$	47.1/55.4	48.4/66.3	28.6	85.8	68.7	120.4
6	+ $\mathcal{L}_{\text{loc}} + \mathcal{L}_{\text{intra}} + \mathcal{L}_{\text{inter}} + \mathcal{L}_{\text{mlm}}$	47.3/55.5	48.5/66.5	29.0	86.3	70.7	122.1

Table 4: Pre-training losses on Tiny-scale model. Involving intra-image region-word alignment loss $\mathcal{L}_{\text{intra}}$, inter-image region-word contrastive loss $\mathcal{L}_{\text{inter}}$ and MLM loss $\mathcal{L}_{\text{mlm}}$ will benefit both localization and understanding tasks.

$\mathcal{L}_{\text{inter}}$	Pre-train Data	COCO	ODinW	LVIS	Flick30K
✗	O365, GoldG	48.06	43.14	25.6	84.36
✓	O365, GoldG	48.59	42.64	26.9	83.90
✗	O365, GoldG, Cap4M	48.21	51.35	34.2	85.56
✓	O365, GoldG, Cap4M	48.79	52.70	35.0	85.50
✗	O365, GoldG, Cap12M	48.50	49.32	35.5	85.79
✓	O365, GoldG, Cap12M	49.26	53.15	36.6	85.84

Table 5: Pre-train data scale up on Base-scale model. Results are reported at the last checkpoint. See supplementary for results at all checkpoints.

Model	COCO Caption			Flickr30K Grounding		
	B4	CIDEr	SPICE	R@1	R@5	R@10
GLIPv2-T	36.5	119.8	21.6	80.8	94.4	96.5
GLIPv2-B	37.4	123.0	21.9	81.0	94.5	96.5

Table 6: GLIPv2 can perform captioning and grounding at the same time (a.k.a., grounded VL understanding).

loss will further boost the model to learn better representation. For more detailed scaling-up effects on various tasks under all the checkpoints for GLIP and GLIPv2, refer to Appendix.

Note that the $(\text{Img}, \text{Text}, T)$ data used in GLIPv2 pre-training can be just human-annotated data (Row1&2 in Table 5), with which GLIPv2 pre-training does not involve any pseudo data from a pre-trained grounding/localization model. In order to achieve the best performance, GLIPv2 uses image-text pair data with pseudo boxes (Cap) from a pre-trained GLIP model (Row3-6 in Table 4), which is trained with the same "grounded VL understanding" task but just with smaller data.

Grounded Vision-Language Understanding GLIPv2 can be trained to perform a VL task and grounding at the same time (Section 3.3). We denote such an ability as grounded VL understanding. In Figure 1, we showcase grounded predictions of GLIPv2 on VQA and COCO captions. We also conduct quantitative evaluations (Table 6). The model achieves strong performance for both VL understanding (on COCO Caption) and localization (on Flickr30K Grounding). Such an ability to produce high-level semantic outputs (i.e., answers and captions) and supporting localization results is another appealing trait of GLIPv2, as potential users can have a better understanding of the model behaviour. See more detailed analysis and qualitative examples in the Appendix.

5 Conclusion and Social Impacts

This paper proposes GLIPv2, a unified framework for VL representation learning that serves both localization tasks and VL understanding tasks. We experimentally verify the effectiveness of the unified model and the novel region-word contrastive learning. Compared to existing methods, GLIPv2 achieves competitive near SoTA performance on various localization and understanding tasks. However, additional analysis of the data and the model is necessary before deploying it in practice since large-scale web data may contain unintended private information, unsuitable images/text, or some bias leakage. Further investigation may be needed for web data due to the above issues.

6 Acknowledgement

We thank anonymous reviewers for their comments and suggestions. Additional thanks go to the Microsoft Research Horizontal AI Team and Microsoft Alexander Multi-modal Team for providing computer resources for large-scale training. The baseline models used in our experiments are based on the open-source code released in the GitHub repository; we acknowledge all the authors who made their code public, which tremendously accelerates our project progress.

References

[1] 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: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 6077–6086. IEEE (2018)

[2] 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: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 6077–6086 (2018)

[3] Antol, S., Agrawal, A., Lu, J., Mitchell, M., Batra, D., Zitnick, C.L., Parikh, D.: VQA: Visual Question Answering. In: International Conference on Computer Vision (ICCV) (2015)

[4] Bansal, A., Sikka, K., Sharma, G., Chellappa, R., Divakaran, A.: Zero-shot object detection. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 384–400 (2018)

[5] Bilen, H., Vedaldi, A.: Weakly supervised deep detection networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2846–2854 (2016)

[6] Bommasani, R., Hudson, D.A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M.S., Bohg, J., Bosselut, A., Brunskill, E., et al.: On the opportunities and risks of foundation models. arXiv preprint arXiv:2108.07258 (2021)

[7] Bucher, M., Vu, T.H., Cord, M., Pérez, P.: Zero-shot semantic segmentation. Advances in Neural Information Processing Systems **32** (2019)

[8] Caesar, H., Uijlings, J., Ferrari, V.: Coco-stuff: Thing and stuff classes in context. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1209–1218 (2018)

[9] Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: European Conference on Computer Vision. pp. 213–229. Springer (2020)

[10] Chen, K., Pang, J., Wang, J., Xiong, Y., Li, X., Sun, S., Feng, W., Liu, Z., Shi, J., Ouyang, W., et al.: Hybrid task cascade for instance segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4974–4983 (2019)

[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)

[12] Chen, Y.C., Li, L., Yu, L., El Kholy, A., Ahmed, F., Gan, Z., Cheng, Y., Liu, J.: UNITER: Universal image-text representation learning. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 104–120. Springer (2020)

[13] Chen, Y.C., Li, L., Yu, L., Kholy, A.E., Ahmed, F., Gan, Z., Cheng, Y., Liu, J.: Uniter: Learning universal image-text representations. arXiv preprint arXiv:1909.11740 (2019)

[14] Dai, J., Li, Y., He, K., Sun, J.: R-fcn: Object detection via region-based fully convolutional networks. In: Advances in neural information processing systems. pp. 379–387 (2016)

[15] Dai, X., Chen, Y., Xiao, B., Chen, D., Liu, M., Yuan, L., Zhang, L.: Dynamic head: Unifying object detection heads with attentions. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7373–7382 (2021)

[16] Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: A Large-Scale Hierarchical Image Database. In: Proceedings of the IEEE conference on computer vision and pattern recognition (2009)

[17] Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)

[18] Gao, P., Geng, S., Zhang, R., Ma, T., Fang, R., Zhang, Y., Li, H., Qiao, Y.: Clip-adapter: Better vision-language models with feature adapters. arXiv preprint arXiv:2110.04544 (2021)

[19] Gokberk Cinbis, R., Verbeek, J., Schmid, C.: Multi-fold mil training for weakly supervised object localization. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2409–2416 (2014)

[20] Goyal, Y., Khot, T., Summers-Stay, D., Batra, D., Parikh, D.: Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 6904–6913 (2017)

[21] Gu, X., Lin, T.Y., Kuo, W., Cui, Y.: Open-vocabulary object detection via vision and language knowledge distillation. arXiv preprint arXiv:2104.13921 (2021)

[22] Gu, X., Lin, T.Y., Kuo, W., Cui, Y.: Zero-shot detection via vision and language knowledge distillation. arXiv preprint arXiv:2104.13921 (2021)

[23] Gupta, A., Dollar, P., Girshick, R.: Lvis: A dataset for large vocabulary instance segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5356–5364 (2019)

[24] Gupta, T., Kamath, A., Kembhavi, A., Hoiem, D.: Towards general purpose vision systems. arXiv preprint arXiv:2104.00743 (2021)

[25] He, K., Chen, X., Xie, S., Li, Y., Dollár, P., Girshick, R.: Masked autoencoders are scalable vision learners. arXiv preprint arXiv:2111.06377 (2021)

[26] He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: Proceedings of the IEEE international conference on computer vision. pp. 2961–2969 (2017)

[27] Hendrycks, D., Gimpel, K.: Gaussian error linear units (gelus). arXiv preprint arXiv:1606.08415 (2016)

[28] Hu, R., Singh, A.: Unit: Multimodal multitask learning with a unified transformer. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 1439–1449 (2021)

[29] Hudson, D.A., Manning, C.D.: Gqa: A new dataset for real-world visual reasoning and compositional question answering. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 6700–6709 (2019)

[30] Kamath, A., Singh, M., LeCun, Y., Synnaeve, G., Misra, I., Carion, N.: Mdetr-modulated detection for end-to-end multi-modal understanding. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 1780–1790 (2021)

[31] Karpathy, A., Joulin, A., Fei-Fei, L.F.: Deep fragment embeddings for bidirectional image sentence mapping. Advances in neural information processing systems **27** (2014)

[32] Kiros, R., Salakhutdinov, R., Zemel, R.S.: Unifying visual-semantic embeddings with multimodal neural language models. arXiv preprint arXiv:1411.2539 (2014)

[33] Krasin, I., Duerig, T., Alldrin, N., Ferrari, V., Abu-El-Haija, S., Kuznetsova, A., Rom, H., Uijlings, J., Popov, S., Veit, A., et al.: Openimages: A public dataset for large-scale multi-label and multi-class image classification. Dataset available from https://github. com/openimages **2**(3), 18 (2017)

[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. International Journal of Computer Vision (IJCV) **123**(1), 32–73 (2017)

[35] Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems **25**, 1097–1105 (2012)

[36] Lester, B., Al-Rfou, R., Constant, N.: The power of scale for parameter-efficient prompt tuning. arXiv preprint arXiv:2104.08691 (2021)

[37] Li, G., Duan, N., Fang, Y., Jiang, D., Zhou, M.: Unicoder-VL: A universal encoder for vision and language by cross-modal pre-training. arXiv preprint arXiv:1908.06066 (2019)

[38] Li, J., Selvaraju, R., Gotmare, A., Joty, S., Xiong, C., Hoi, S.C.H.: Align before fuse: Vision and language representation learning with momentum distillation. Advances in Neural Information Processing Systems **34** (2021)

[39] 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)

[40] Li, L.H., You, H., Wang, Z., Zareian, A., Chang, S.F., Chang, K.W.: Unsupervised vision-and-language pre-training without parallel images and captions. arXiv preprint arXiv:2010.12831 (2020)

[41] Li, L.H., Zhang, P., Zhang, H., Yang, J., Li, C., Zhong, Y., Wang, L., Yuan, L., Zhang, L., Hwang, J.N., et al.: Grounded language-image pre-training. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10965–10975 (2022)

[42] Li, X., Yin, X., Li, C., Zhang, P., Hu, X., Zhang, L., Wang, L., Hu, H., Dong, L., Wei, F., et al.: Oscar: Object-semantics aligned pre-training for vision-language tasks. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 121–137. Springer (2020)

[43] Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. In: Proceedings of the IEEE international conference on computer vision. pp. 2980–2988 (2017)

[44] 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: European conference on computer vision. pp. 740–755. Springer (2014)

[45] Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. arXiv preprint arXiv:2103.14030 (2021)

[46] Lu, J., Batra, D., Parikh, D., Lee, S.: ViLBERT: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In: Advances in Neural Information Processing Systems (NeurIPS). pp. 13–23 (2019)

[47] Lu, J., Clark, C., Zellers, R., Mottaghi, R., Kembhavi, A.: Unified-io: A unified model for vision, language, and multi-modal tasks. arXiv preprint arXiv:2206.08916 (2022)

[48] Ma, C.Y., Kalantidis, Y., AlRegib, G., Vajda, P., Rohrbach, M., Kira, Z.: Learning to generate grounded visual captions without localization supervision. In: European Conference on Computer Vision. pp. 353–370. Springer (2020)

[49] Newell, A., Yang, K., Deng, J.: Stacked hourglass networks for human pose estimation. In: European conference on computer vision. pp. 483–499. Springer (2016)

[50] Plummer, B.A., Wang, L., Cervantes, C.M., Caicedo, J.C., Hockenmaier, J., Lazebnik, S.: Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In: Proceedings of the IEEE international conference on computer vision. pp. 2641–2649 (2015)

[51] Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International Conference on Machine Learning (ICML) (2021)

[52] Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 779–788 (2016)

[53] Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in neural information processing systems **28**, 91–99 (2015)

[54] Shao, S., Li, Z., Zhang, T., Peng, C., Yu, G., Zhang, X., Li, J., Sun, J.: Objects365: A large-scale, high-quality dataset for object detection. In: Proceedings of the IEEE international conference on computer vision. pp. 8430–8439 (2019)

[55] Shin, T., Razeghi, Y., Logan IV, R.L., Wallace, E., Singh, S.: Autoprompt: Eliciting knowledge from language models with automatically generated prompts. arXiv preprint arXiv:2010.15980 (2020)

[56] Su, W., Zhu, X., Cao, Y., Li, B., Lu, L., Wei, F., Dai, J.: VL-BERT: Pre-training of generic visual-linguistic representations. arXiv preprint arXiv:1908.08530 (2019)

[57] Tan, H., Bansal, M.: Lxmert: Learning cross-modality encoder representations from transformers. arXiv preprint arXiv:1908.07490 (2019)

[58] Teney, D., Anderson, P., He, X., Van Den Hengel, A.: Tips and tricks for visual question answering: Learnings from the 2017 challenge. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4223–4232 (2018)

[59] Tian, Z., Shen, C., Chen, H., He, T.: Fcos: Fully convolutional one-stage object detection. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 9627–9636 (2019)

[60] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: Advances in neural information processing systems. pp. 5998–6008 (2017)

[61] Wang, P., Cai, Z., Yang, H., Swaminathan, G., Vasconcelos, N., Schiele, B., Soatto, S.: Omni-detr: Omni-supervised object detection with transformers. arXiv preprint arXiv:2203.16089 (2022)

[62] Wang, X., Huang, T.E., Darrell, T., Gonzalez, J.E., Yu, F.: Frustratingly simple few-shot object detection. arXiv preprint arXiv:2003.06957 (2020)

[63] Wu, C., Lin, Z., Cohen, S., Bui, T., Maji, S.: Phrasecut: Language-based image segmentation in the wild. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10216–10225 (2020)

[64] Xiong, R., Yang, Y., He, D., Zheng, K., Zheng, S., Xing, C., Zhang, H., Lan, Y., Wang, L., Liu, T.: On layer normalization in the transformer architecture. In: International Conference on Machine Learning. pp. 10524–10533. PMLR (2020)

[65] Yang, J., Li, C., Zhang, P., Xiao, B., Liu, C., Yuan, L., Gao, J.: Unified contrastive learning in image-text-label space. arXiv preprint arXiv:2204.03610 (2022)

[66] Yang, Z., Gan, Z., Wang, J., Hu, X., Ahmed, F., Liu, Z., Lu, Y., Wang, L.: Crossing the format boundary of text and boxes: Towards unified vision-language modeling. arXiv preprint arXiv:2111.12085 (2021)

[67] Yuan, L., Chen, D., Chen, Y.L., Codella, N., Dai, X., Gao, J., Hu, H., Huang, X., Li, B., Li, C., et al.: Florence: A new foundation model for computer vision. arXiv preprint arXiv:2111.11432 (2021)

[68] Zareian, A., Rosa, K.D., Hu, D.H., Chang, S.F.: Open-vocabulary object detection using captions. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14393–14402 (2021)

[69] Zeng, Y., Zhang, X., Li, H.: Multi-grained vision language pre-training: Aligning texts with visual concepts. arXiv preprint arXiv:2111.08276 (2021)

[70] 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: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5579–5588 (2021)

[71] 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: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 5579–5588 (June 2021)

[72] Zhong, Y., Yang, J., Zhang, P., Li, C., Codella, N., Li, L.H., Zhou, L., Dai, X., Yuan, L., Li, Y., et al.: Regionclip: Region-based language-image pretraining. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16793–16803 (2022)

[73] Zhou, K., Yang, J., Loy, C.C., Liu, Z.: Learning to prompt for vision-language models. arXiv preprint arXiv:2109.01134 (2021)

[74] Zhou, L., Palangi, H., Zhang, L., Hu, H., Corso, J., Gao, J.: Unified vision-language pre-training for image captioning and vqa. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 34, pp. 13041–13049 (2020)

[75] Zhou, L., Palangi, H., Zhang, L., Hu, H., Corso, J.J., Gao, J.: Unified vision-language pre-training for image captioning and VQA. AAAI (2020)

[76] Zhu, P., Wang, H., Saligrama, V.: Zero shot detection. IEEE Transactions on Circuits and Systems for Video Technology **30**(4), 998–1010 (2019)

[77] Zhu, P., Wang, H., Saligrama, V.: Don't even look once: Synthesizing features for zero-shot detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11693–11702 (2020)

Checklist

The checklist follows the references. Please read the checklist guidelines carefully for information on how to answer these questions. For each question, change the default **[TODO]** to [Yes] , [No] , or

[N/A] . You are strongly encouraged to include a **justification to your answer**, either by referencing the appropriate section of your paper or providing a brief inline description. For example:

- Did you include the license to the code and datasets? [Yes]
- Did you include the license to the code and datasets? [No] The code and the data are proprietary.
- Did you include the license to the code and datasets? [N/A]

Please do not modify the questions and only use the provided macros for your answers. Note that the Checklist section does not count towards the page limit. In your paper, please delete this instructions block and only keep the Checklist section heading above along with the questions/answers below.

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [Yes]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [Yes]
   (c) Did you include any new assets either in the supplemental material or as a URL? [No]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

Appendix

The appendix is organized as follows:

- In Section A, we provide more visualizations of our model's predictions on various localization and VL understanding tasks.
- In Section B, we describe all our evaluated tasks and their dataset in detail.
- In Section C, we discuss the difference between our additional inter-image region-word contrastive loss and some other well-known losses that were also applied over a full batch in multiple works.
- In Section D, we introduce the training details and hyperparameters used in Section 4 in the main paper.
- Section E, we analyze the effect of using different language encoder and their pre-trained weights in our models.
- In Section F, we provide more results for all the checkpoints of adding pre-training data (refer to Section 4 in the main paper).
- In Section G, we provide a detailed analysis of the experiments of grounded captioning (mentioned in Section 4 in the main paper).
- In Section H, we give out a comparison for the model's inference speed.
- In Section I, we clearly provide the original sources of the images that are used in our paper.
- In Section J, we present per-dataset results for all experiments in ODinW.

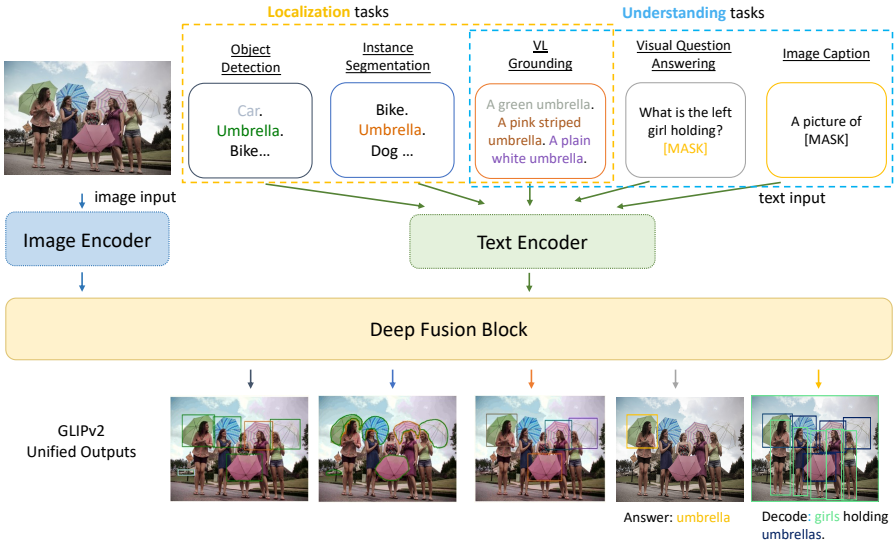

Figure 4: GLIPv2, a pre-trained grounded VL understanding model, unifies various localization and VL understanding tasks. These two kinds of tasks mutually benefit each other and enable new capabilities such as language-guided detection/segmentation and grounded VQA/captioning.

A Visualization

We provide a clearer illustration of GLIPv2 in Figure 4, which elegantly unifies various localization (object detection, instance segmentation) and VL understanding (phrase grounding, VQA and captioning) tasks. More visualizations of the predictions under various tasks from GLIPv2 are also provided to indicate the model's strength and capability. Please refer to Figure 5 for OD / Grounding, Figure 6 for Instance / Referring Image Segmentation, and Figure 7 for Grounded VL Understanding.

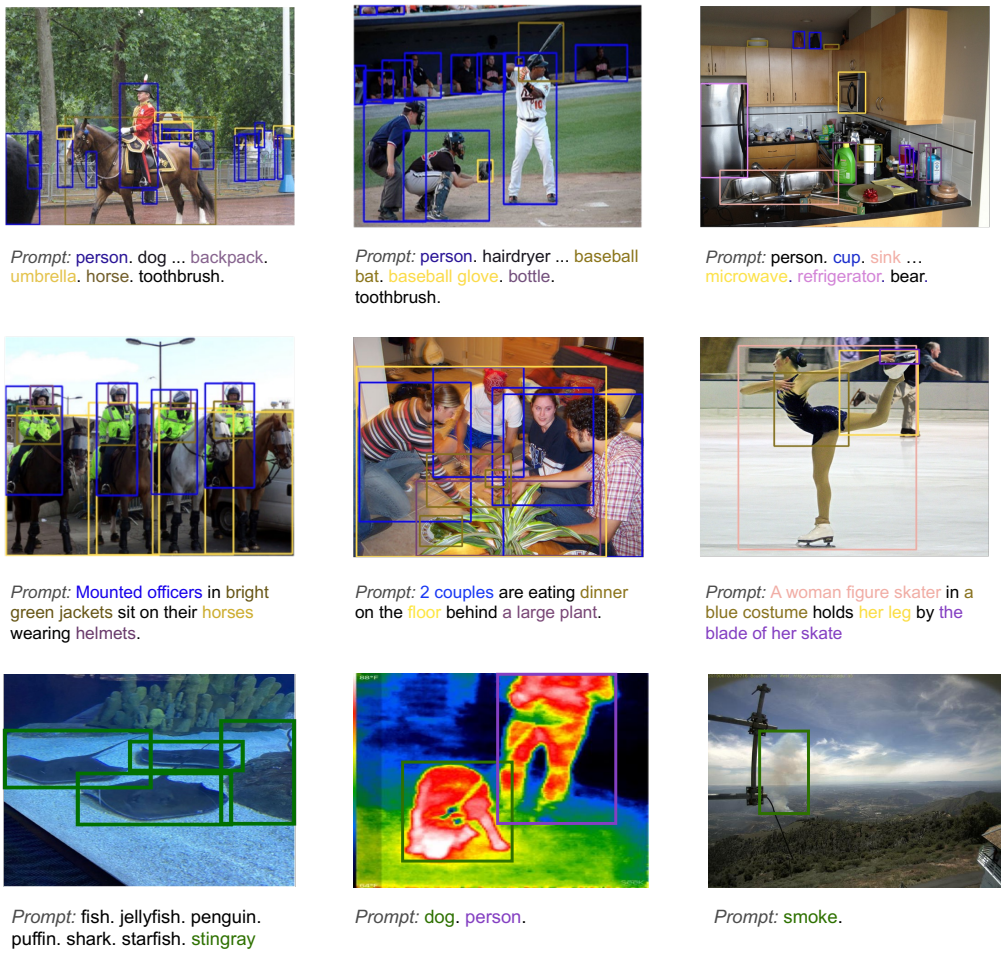

Figure 5: Visualization for OD / Grounding. Row 1: Object Detection on COCO. Row 2: Phrase Grounding on Flickr30K. Row 3: Object Detection on ODinW.

B Tasks and dataset descriptions

B.1 (Language-guided) object detection and phrase grounding

COCO. [8] The Microsoft Common Objects in Context dataset is a medium-scale object detection dataset. It has about 900k bounding box annotations for 80 object categories, with about 7.3 annotations per image. It is one of the most used object detection datasets, and its images are often used within other datasets (including VG and LVIS).

Flickr30k-entities. [50] Given one or more phrases, which may be interrelated, the phrase grounding task is to provide a set of bounding boxes for each given phrase. We use the Flickr30k-entities dataset for this task, with the train/val/test splits as provided by [41] and evaluate our performance in terms of Recall. Flickr30K is included in the gold grounding data so we directly evaluate the models after pre-training as in MDETR [30]. We predict use any-box protocol specified in MDETR.

ODinW. We use 13 datasets from Roboflow[2]. Roboflow hosts over 30 datasets, and we exclude datasets that are too challenging (e.g., detecting different kinds of chess pieces) or impossible to solve without specific domain knowledge (e.g., understanding sign language). We provide the details of the 13 datasets we use in Table 7. We include the PASCAL VOC 2012 dataset as a reference dataset, as public baselines have been established on this dataset. For PascalVOC, we follow the convention

[2] `https://public.roboflow.com/object-detection`

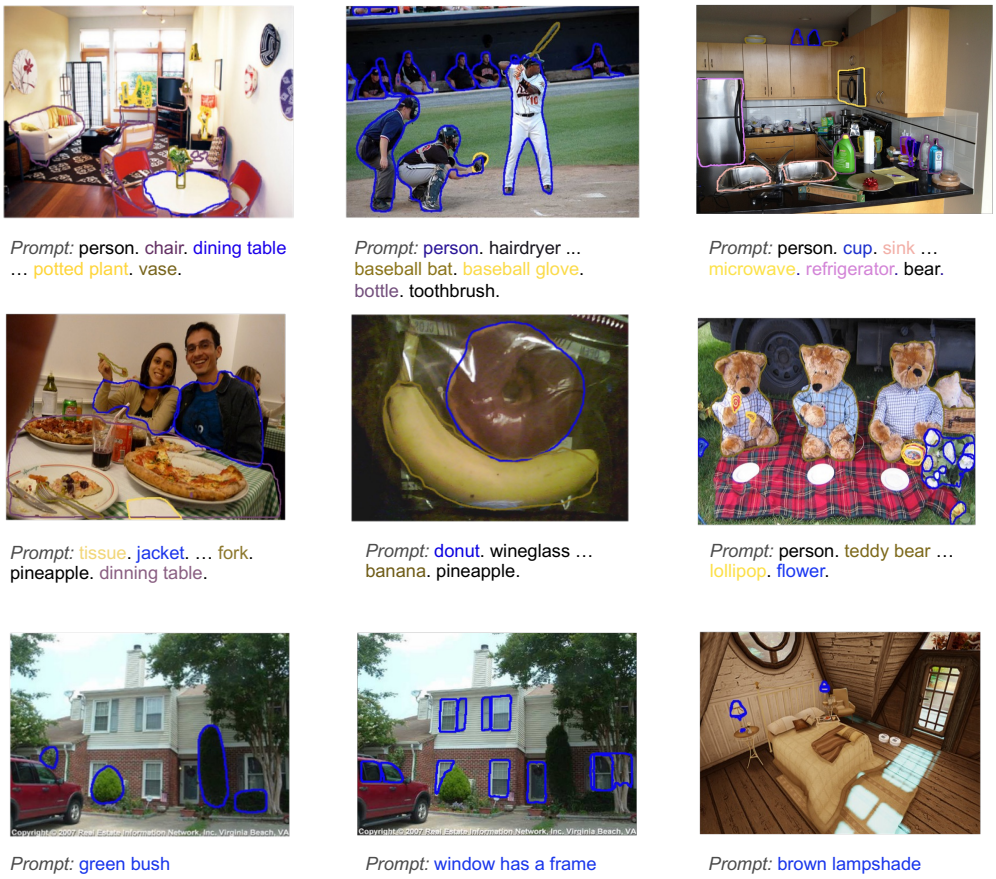

Figure 6: Visualization for Instance / Referring Image Segmentation. Row 1: Instance Segmentation on COCO Mask. Row 2: Instance Segmentation on LVIS. Row 3: Referring Image Segmentation on PhraseCut.

Dataset	Objects of Interest	Train/Val/Test	URL
PascalVOC	Common objects (PascalVOC 2012)	13690/3422/-	https://public.roboflow.com/object-detection/pascal-voc-2012
AerialDrone	Boats, cars, etc. from drone images	52/15/7	https://public.roboflow.com/object-detection/aerial-maritime
Aquarium	Penguins, starfish, etc. in an aquarium	448/127/63	https://public.roboflow.com/object-detection/aquarium
Rabbits	Cottontail rabbits	1980/19/10	https://public.roboflow.com/object-detection/cottontail-rabbits-video-dataset
EgoHands	Hands in ego-centric images	3840/480/480	https://public.roboflow.com/object-detection/hands
Mushrooms	Two kinds of mushrooms	41/5/5	https://public.roboflow.com/object-detection/na-mushrooms
Packages	Delivery packages	19/4/3	https://public.roboflow.com/object-detection/packages-dataset
Raccoon	Raccoon	150/29/17	https://public.roboflow.com/object-detection/raccoon
Shellfish	Shrimp, lobster, and crab	406/116/58	https://public.roboflow.com/object-detection/shellfish-openimages
Vehicles	Car, bus, motorcycle, truck, and ambulance	878/250/126	https://public.roboflow.com/object-detection/vehicles-openimages
Pistols	Pistol	2377/297/297	https://public.roboflow.com/object-detection/pistols/1
Pothole	Potholes on the road	465/133/67	https://public.roboflow.com/object-detection/pothole
Thermal	Dogs and people in thermal images	142/41/20	https://public.roboflow.com/object-detection/thermal-dogs-and-people

Table 7: 13 ODinW dataset statistics. We summarize the objects of interest for each dataset and report the image number of each split.

and report on the validation set. For Pistols, there are no official validation or test sets so we split the dataset ourselves.

B.2 (Language-guided) instance segmentation and referring image segmentation

LVIS. [23] The Large Vocabulary Instance Segmentation dataset has over a thousand object categories, following a long-tail distribution with some categories having only a few examples. Similar to VG, LVIS uses the same images as in COCO, re-annotated with more object categories. In contrast to COCO, LVIS is a federated dataset, which means that only a subset of categories is annotated in each image. Annotations, therefore, include positive and negative object labels for objects that are present

Figure 7: Visualization for Grounded VL Understanding. Row 1: Grounded VQA predictions (The model is given the input question and a placeholder token "[MASK]" for the answer. The model can ground not only entities in the question but also the implied answer entity). Row 2: Grounded captioning on COCO (The model can generate high-quality captions and, in the meantime, provide localization results.

and categories that are not present, respectively. In addition, LVIS categories are not pairwise disjoint, such that the same object can belong to several categories.

PhraseCut. [63] Besides object detection, we show that our GLIPv2 can be extended to perform segmentation by evaluating the referring expression segmentation task of the recent PhraseCut[63] which consists of images from VG, annotated with segmentation masks for each referring expression. These expressions comprise a wide vocabulary of objects, attributes and relations, making it a challenging benchmark. Contrary to other referring expression segmentation datasets, in PhraseCut the expression may refer to several objects and the model is expected to find all the corresponding instances.

B.3 VQA and image captioning

VQA. [20] requires the model to predict an answer given an image and a question. We conduct experiments on the VQA2.0 dataset, which is constructed using images from COCO. It contains 83k images for training, 41k for validation, and 81k for testing. We treat VQA as a classification problem with an answer set of 3,129 candidates following the common practice of this task. For our best models, we report test-dev and test-std scores by submitting to the official evaluation server.[3]

COCO image captioning. [11] The goal of image captioning is to generate a natural language description given an input image. We evaluate GLIPv2 on COCO Captioning dataset and report BLEU-4, CIDEr, and SPICE scores on the Karparthy test split.

C Difference between inter-image region-word contrastive loss with other "region-word" losses.

As far as we know, up to the deadline (05/19/2022) for NeurIPS submission, there are only three published papers (VILD [21], RegionCLIP [72], and X-VLM [69]) that have the flavor of "region-

[3] `https://eval.ai/challenge/830/overview`

Model	Image	Text	Pre-Train Data		
			Detection	Grounding	Caption
GLIPv2-T	Swin-T	BERT-Base	O365	GoldG (no COCO)	Cap4M
GLIPv2-B	Swin-B	CLIP	O365, COCO, OpenImages, VG, ImageNetBoxes	GoldG	CC15M+ SBU
GLIPv2-H	CoSwin-H [67]	CLIP	O365, COCO, OpenImages, VG, ImageNetBoxes	GoldG	CC15M+SBU
Mask Head	–	–	LVIS, COCO	PhraseCut	–

Table 8: A detailed list of GLIPv2 model variants

word" loss applied over full batch. We discuss the difference between our work and the three aforementioned works in the following:

1. All these three works use “region-sentence" loss, i.e., the similarity between a region feature and the [CLS] token of a sentence, instead of true "region-word" loss used in GLIPv2. As a result, none of these three works made use of the phrase grounding data, which may contain multiple entities in one sentence during their training. It is the most important point in GLIPv2 to use phrase grounding data and pseudo grounding data to train a unified grounded VL understanding model.
2. GLIPv2 has carefully designed the positive label propagation in our inter-image region-word contrastive loss to mitigate the wrong assumption that "every unpaired region-word pair is negative". As far as we know, no previous work has mentioned this mechanism of positive label propagation before.
3. There are some other differences. For example, in VILD, its “region-sentence loss" is actually not a contrastive loss over full-batch but a classification loss over a fixed vocabulary per sample (see the definition of $L_{ViLD-text}$).

Upon all three points above, we believe that our inter-image region-word contrastive loss is novel and has a significant difference from previous works.

D Training details and hyperparamters

D.1 Pre-training

Pre-training data. There are three different types of data in pre-training 1) detection data 2) grounding data 3) caption data, as shown in Table 8. The detection data includes Object365 [54], COCO [8], OpenImages [33], Visual Genome [34], and ImageNetBoxes [16]. The grounding data includes GoldG, 0.8M human-annotated gold grounding data curated by MDETR [30] combining Flick30K, VG Caption, and GQA [29]. The Cap4M is a 4M image-text pairs collected from the web with boxes generated by GLIP-T(C) in [41], and CC (Conceptual Captions) + SBU (with 1M data).

Implementation details. In Section 4 in the main paper, we introduced GLIPv2-T, GLIPv2-B, GLIPv2-H, and we introduce the implementation details in the following.

We pre-train GLIPv2-T based on Swin-Tiny models with 32 GPUs and a batch size of 64. We use a base learning rate of 1×10^{-5} for the language backbone (BERT-Base) and 1×10^{-4} for all other parameters. The learning rate is stepped down by a factor of 0.1 at the 67% and 89% of the total 330,000 training steps. We decay the learning rate when the zero-shot performance on COCO saturates. The max input length is 256 tokens for all models. To optimize the results for object detection, we continue pre-training without the MLM loss for another 300,000 steps.

We pre-train GLIPv2-B based on Swin-Base models with 64 GPUs and a batch size of 64. We use a base learning rate of 1×10^{-4} for all parameters, including the language backbone (CLIP-type pre-layernorm transformer). The learning rate is stepped down by a factor of 0.1 at the 67% and 89% of the total 1 million training steps. We decay the learning rate when the zero-shot performance on COCO saturates. The max input length is 256 tokens for all models. To optimize the results for object detection, we continue pre-training without the MLM loss for another 500,000 steps.

We pre-train GLIPv2-H based on the CoSwin-Huge model from Florence [67] with 64 GPUs and a batch size of 64. We use a base learning rate of 1×10^{-4} for all parameters, including the language backbone (CLIP-type pre-layernorm transformer). The learning rate is stepped down by a factor of

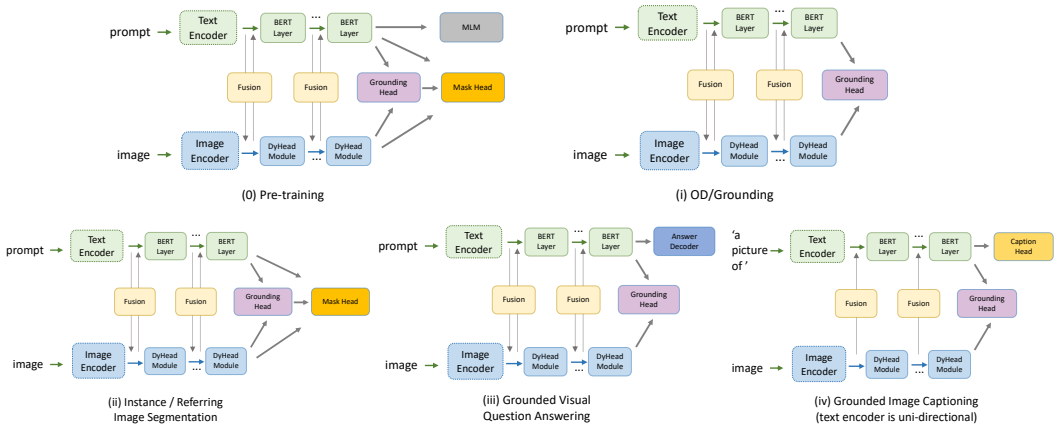

Figure 8: The model architecture for pre-training (0), and downstream tasks (i) OD / Grounding (ii) Instance / Referring Image Segmentation (iii) Grounded Visual Question Answering (iv) Grounded Image Captioning.

0.1 at the 67% and 89% of the total 1 million training steps. We decay the learning rate when the zero-shot performance on COCO saturates. The max input length is 256 tokens for all models. We found that there is **no** need to continue pre-training without MLM loss for the huge model.

Mask heads of GLIPv2-T, GLIPv2-B and GLIPv2-H are pre-trained COCO, LVIS and PhraseCut, while freezing all the other model parameters. This mask head pre-training uses batch size 64, and goes through COCO for 24 epochs, LVIS for 24 epochs, and PhraseCut for 8 epochs, respectively. GLIPv2 uses Hourglass network [49] as instance segmentation head feature extractor, and utilizes the "classification-to-matching" trick to change the instance segmentation head linear prediction layer (outputs K-dimensional logits on each pixel) to a dot product layer between pixel visual features and the word features after VL fusion. GLIPv2-T and GLIPv2-B use a very basic Hourglass network for segmentation head feature extractor: only 1 scale and 1 layer, with hidden dimension 256. GLIPv2-H uses a larger Hourglass network for segmentation head feature extractor: 2 scales and 4 layers, with hidden dimension 384.

D.2 Downstream tasks

OD / Grounding. When fine-tuning on COCO, we use a base learning rate of 1×10^{-5} and 24 training epochs for the pre-trained GLIPv2-T model, and a base learning rate of 5×10^{-6} and 5 training epochs for the pre-trained GLIPv2-B and GLIPv2-H models.

For direct evaluation on LVIS, since LVIS has over 1,200 categories and they cannot be fit into one text prompt, so we segment them into multiple chunks, fitting 40 categories into one prompt and query the model multiple times with the different prompts. We find that models tend to overfit on LVIS during the course of pre-training so we closely monitor the performance on minival for all models and report the results with the best checkpoints in Table 2 in the main paper.

For direct evaluation on Flickr30K, models may also overfit during the course of pre-training so we monitor the performance on the validation set for all models and report the results with the best checkpoints in Table 2 in the main paper.

Instance segmentation / Referring Image Segmentation. Given the pre-trained model with pre-trained mask head, we simply fine-tune the **entire** network to get the task-specific fine-tuned models.

For fine-tuning on COCO instance segmentation, we use a base learning rate of 1×10^{-5} and 24 training epochs for the pre-trained GLIPv2-T model, and a base learning rate of 5×10^{-6} and 5 training epochs for the pre-trained GLIPv2-B and GLIPv2-H models.

For fine-tuning on LVIS instance segmentation, we use a base learning rate of 1×10^{-5} and 24 training epochs for the pre-trained GLIPv2-T model, and a base learning rate of 5×10^{-6} and 5 training epochs for the pre-trained GLIPv2-B and GLIPv2-H models.

For fine-tuning on PhraseCut Referring Image segmentation, we use a base learning rate of 1×10^{-5} and 12 training epochs for the pre-trained GLIPv2-T model, and a base learning rate of 5×10^{-6} and 3 training epochs for the pre-trained GLIPv2-B and GLIPv2-H models.

(Grounded) VQA. To fine-tune GLIPv2 for VQA, we feed the image and question into the model and then take the output feature sequence P from the language side (after the VL fusion) and apply a 'attention pooling' layer to obtain a feature vector P_{vqa}. More specifically, the attention pooling layer applies a linear layer followed by softmax to obtain normalized scaler weights, and then these weights are used to compute a weighted sum to produce the feature vector p_{vqa}. This feature vector is then fed to a 2-layer MLP with GeLU activation [27] and a final linear layer to obtain the logits for the 3129-way classification.[4] Following standard practice [58], we use binary cross entropy loss to take account of different answers from multiple human annotators. Following VinVL [71], we train on the combination of train2014 + val2014 splits of the VQAv2 dataset, except for the reserved 2k dev split.[5]. For the ablation studies we report the accuracy on this 2k dev split.

Other than the conventional VQA setting, we also experimented a new 'grounded VQA' setup, which the model is required to not only predict the answer, but also ground the objects (predict bounding boxes in the image) mentioned in the question and answer text, see Figure 8(iii). Note that the language input is the question appended by a `[MASK]` token, and this `[MASK]` token should ground to the object if the answer is indeed an object in the image. The total training loss is summing the grounding loss (intra-image region-word contrastive loss) and the VQA loss described previously.

(Grounded) Image Captioning. We fine-tune the pre-trained model on COCO Caption "Karpathy" training split. The training objective is uni-directional Language Modeling (LM), which maximizes the likelihood of the next word at each position given the image and the text sequence before it. To enable autoregressive generation, we use uni-directional attention mask for the text part, and prevent the image part from attending to the text part in the fusion layers. Although the training objective (LM) is different from that in pre-training (i.e., bi-directional MLM), we directly fine-tune the model for image captioning to evaluate its capability of generalizing to VL generation tasks. Our model is trained with cross entropy loss only, without using CIDEr optimization.

For grounded image captioning (Figure 8), we add the grounding loss (intra-image region-word contrastive loss) in training, which is calculated in the same way as in pre-training. We use Flickr30K training split for this task. During inference, for each predicted text token, we get its dot product logits with all the region representations and choose the maximum as the associated bounding box.

E Analysis on the effect of different language encoders and pre-trained weights

For GLIPv2-T, we use the ImageNet pre-trained Swin-Transformer to initialize the image encoder and BERT-base-uncased to initialize the language encoder. For GLIPv2-B, we use the pre-trained paired image-language encoder from UniCL (CLIP-like pre-training, `https://github.com/microsoft/UniCL`) for initialization. We did an ablation study on the different language encoders (UniCL vs. BERT) and found that their results are nearly the same, as shown in Figure 9. Therefore, UniCL initialization does not skew the good localization performance. The main reason for us to keep the UniCL(CLIP-like) language encoder is due to its Pre-LayerNorm [64] operation. We find the UniCL(CLIP-like) language encoder with Pre-LayerNorm is more stable during the training compared with BERT, which uses Post-LayerNorm.

F More analysis on pre-training data

Table 5 in the main paper reports the last checkpoint results on GLIPv2 when we do the scaling up of pre-training data. As more weak image-text pair data (Cap) is involved in our training, it benefits both standard/in-domain (i.e., COCO, Flickr30K) and large-domain gap (i.e., ODinW, LVIS) tasks. Further adding the inter-image region-word contrastive helps when we are fixing the data at the same scale. For large-domain gap tasks, adding the inter-image region-word contrastive loss will further

[4] We experimented simpler pooling methods such as average pooling and `[CLS]` pooling [17] in the early experiments and found the attention pooling described above works better.

[5] 2000 images sampled from the val2014 split (and their corresponding question-answer pairs).

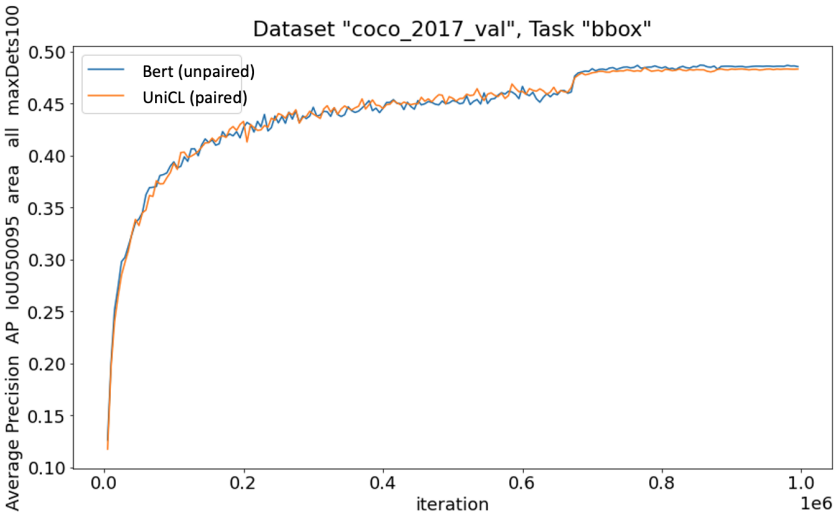

Figure 9: GLIP-B with image encoder initialized from UniCL pre-trained image encoder, but with different language encoder initialization. Blue: language encoder initialized by Bert-Base, thus un-paired image-language pre-trained encoders. Yellow: language encoder initialized from UniCL pre-trained language encoder, thus paired UniCL pre-trained image-language encoders. From the results, we can see that the COCO zero-shot transfer results from two initializations are nearly the same. Similar results have been observed for other metrics, i.e., LVIS zero-shot AP, ODinW benchmark, and Flickr30k grounding performance.

boost the model to learn better representation. To learn more scaling-up effects on various tasks under all the checkpoints for GLIP and GLIPv2, see Figure 10. Given the considerable amount of improvement of GLIPv2 when the number of caption data increases from 0M to 12M, we hypothesize that it has potential to further grow by training on even larger-scale web image-text pairs.

G Experiments on grounded image captioning

The grounded captioning task requires the model to generate an image caption and also ground predicted phrases to object regions. The final predictions consist of (1) the text captions (2) predicted object regions, and (3) the grounding correspondence between the phrases and regions. Following the established benchmarks [48, 74], we evaluate the caption metrics on COCO Captions and report the grounding metrics on Flick30K, as shown in Table 9.

Model	COCO Caption			Flickr30K Grounding		
	B@4	CIDEr	SPICE	R@1	R@5	R@10
No Pretrain	35.4	115.3	21.2	77.0	92.9	95.7
+ L_{mlm}	33.4	107.6	19.9	70.9	90.0	93.2
+ $L_{\text{loc}} + L_{\text{intra}} + L_{\text{inter}}$	36.6	120.3	21.6	80.8	94.9	96.7
GLIPv2-T	36.5	119.8	21.6	80.8	94.4	96.5
GLIPv2-B	37.4	123.0	21.9	81.0	94.5	96.5

Table 9: Grounded image captioning results on the COCO Caption, and Flickr30K Entities. We report BLEU@4, CIDer, and SPICE metrics for caption evaluation, and we use R@1, R@5, R@10 for grounding evaluation.

H Inference speed

We test the inference speed for GLIPv2 on V100 with batch size 1 and show its comparison to MDETR, as shown in Table 10.

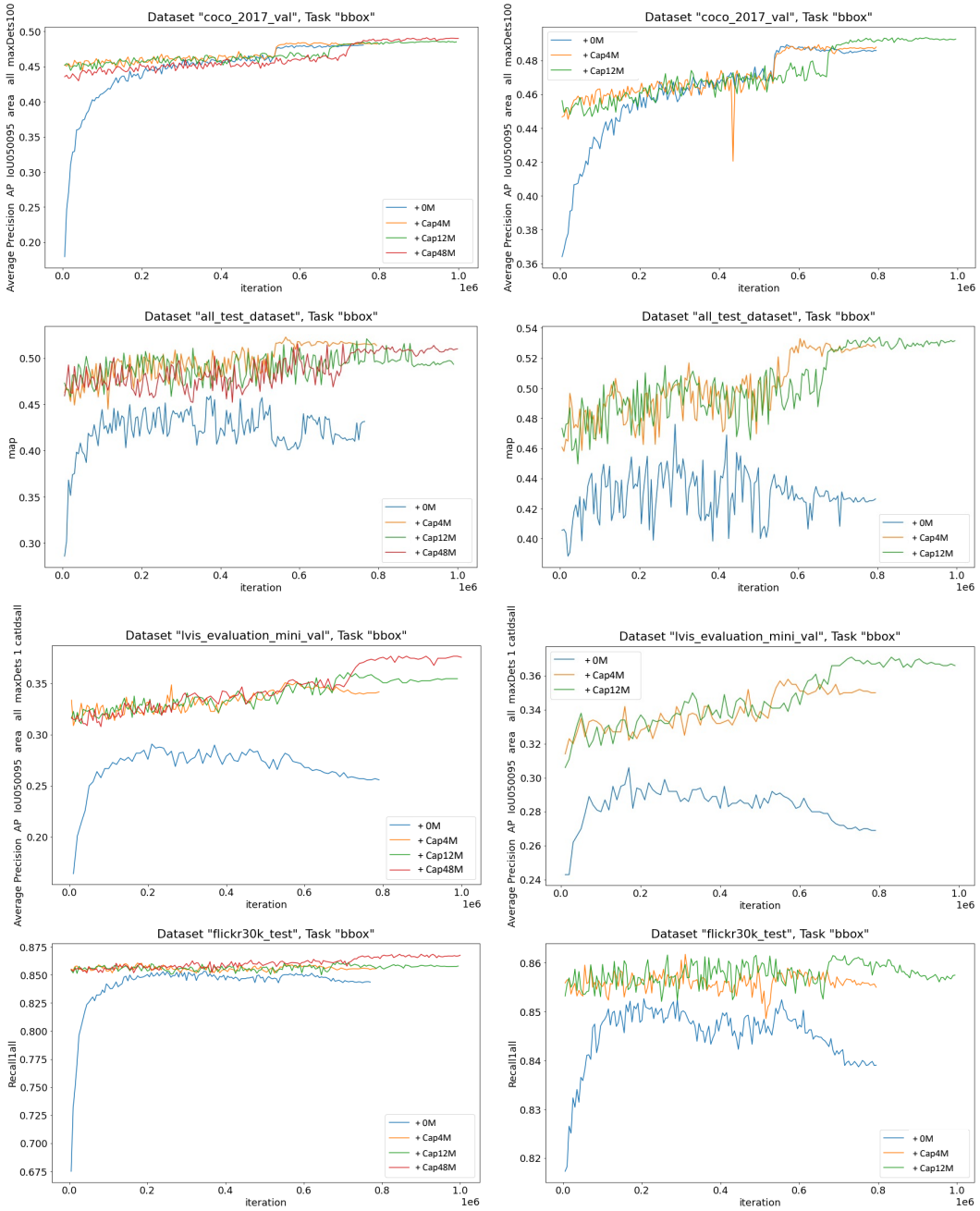

Figure 10: Pre-train data scale up on Base-scale model. Left: GLIP, Right: GLIPv2; Row 1: COCO minival, Row 2: ODinW test split, Row 3: LVIS minival, Row 4: Flick30K test.

I Figure Reference

We provided the original sources of the images that are used in our paper in the following. All datasets above were collected by the creators (cited) and consent for any personally identifiable information (PII) was ascertained by the authors where necessary.

Figure 1 in the main paper - The top left and the bottom middle figures are the 281759.jpg in COCO val set; The left right images are (from top to down: (1) 2588.jpg in ODinW Aquarium test set. (2) 13923.jpg in LVIS val set. (3) 132690.jpg in VQA2.0 val set (question id is 132690002). (4) 462565.jpg in COCO Caption val set.

Model	Object Detection (COCO)	Phrase Grounding (Flick30K)	Referring Expression Segmentation (PhraseCut)
MDETR R101 [30]	–	9.31	3.80
MDETR EffB3 [30]	–	11.20	3.98
MDETR EffB5 [30]	–	9.15	–
GLIPv2-T	4.12	3.74	2.26
GLIPv2-B	3.01	3.23	2.39
GLIPv2-H	1.21	1.13	0.89

Table 10: Model inference speed on various tasks. We report FPS, which is the number of images processed per second per GPU (higher is better).

Figure 2 in the main paper - The top left figure is the 209297.jpg in COCO train set; The bottom left figure is the 9378.jpg in COCO val set.

Figure 4 in the Appendix - Same as Figure 1. The top left and the bottom middle figures are the 281759.jpg in COCO val set.

Figure 5 in the Appendix - Row 1 (from left to right): (1) 439715.jpg in COCO val set. (2) 6471.jpg in COCO val set. (3) 13923.jpg in COCO val set; Row 2: (1) 5521996.jpg in Flickr30K val set. (2) 764507.jpg in Flickr30K val set. (3) 7520721.jpg in Flick30K val set; Row 3: (1) 2588.jpg in ODinW Aquarium test set. (2) 143.jpg in Thermal val set. (3) ck0l9j6n6oqjo0848ps5blk3b.jpg in WildFire val set.

Figure 6 in the Appendix - Row 1 (from left to right): (1) 13923.jpg in COCO val set. (2) 6471.jpg in COCO val set. (3) 7574.jpg in COCO val set; Row 2: (1) 117320.jpg in LVIS val set. (2) 2587.jpg in LVIS val set. (3) 211120.jpg in LVIS val set; Row 3: (1) 4744.jpg in PhraseCut test set. (2) 4744.jpg in PhraseCut val set. (3) 567.jpg in PhraseCut train set.

Figure 7 in the Appendix - Row 1 (from left to right): (1) 486.jpg in VQA2.0 val set (question id is 486002). (2) 262746.jpg in VQA2.0 val set (question id is 262746002). (3) 132690.jpg in VQA2.0 val set (question id is 132690002); Row 2: (1) 391895.jpg in COCO Caption val set. (2) 462565.jpg in COCO Caption val set. (3) 579056.jpg in COCO Caption val set.

J All results for ODinW

We report the per-dataset performance under 0,1,3,5,10-shot and full data as well as prompt tuning, and full-model tuning in Table 11 and Table 12 (on the next page).

Model	PascalVOC	AerialDrone	Aquarium	Rabbits	EgoHands	Mushrooms	Packages	Raccoon	Shellfish	Vehicles	Pistols	Pothole	Thermal	Avg
GLIP-T	56.2	12.5	18.4	70.2	50.0	73.8	72.3	57.8	26.3	56.0	49.6	17.7	44.1	46.5
GLIP-L	61.7	7.1	26.9	75.0	45.5	49.0	62.8	63.3	68.9	57.3	68.6	25.7	66.0	52.1
GLIPv2-T	57.6	10.5	18.4	71.4	52.7	77.7	67.7	58.8	27.8	55.6	60.1	20.0	52.4	48.5
GLIPv2-B	62.8	8.6	18.9	73.7	50.3	83.0	68.6	61.6	56.0	53.8	67.8	32.6	53.8	54.2
GLIPv2-H	66.3	10.9	30.4	74.6	55.1	52.1	71.3	63.8	66.2	57.2	66.4	33.8	73.3	55.5

Table 11: Zero-shot performance on 13 ODinW datasets.

Model	Shot	Tune	PascalVOC	AerialDrone	Aquarium	Rabbits	EgoHands	Mushrooms	Packages	Raccoon	Shellfish	Vehicles	Pistols	Pothole	Thermal	Avg
DyHead O365	1	Full	$25.8_{\pm 3.0}$	$16.5_{\pm 1.8}$	$15.9_{\pm 2.7}$	$55.7_{\pm 6.0}$	$44.0_{\pm 3.6}$	$66.9_{\pm 3.9}$	$54.2_{\pm 5.7}$	$50.7_{\pm 7.7}$	$14.1_{\pm 3.6}$	$33.0_{\pm 11.0}$	$11.0_{\pm 6.5}$	$8.2_{\pm 4.1}$	$43.2_{\pm 10.0}$	$33.8_{\pm 3.5}$
DyHead O365	3	Full	$40.4_{\pm 1.0}$	$20.5_{\pm 4.0}$	$26.5_{\pm 1.3}$	$57.9_{\pm 2.0}$	$53.9_{\pm 2.5}$	$76.5_{\pm 2.3}$	$62.6_{\pm 13.3}$	$52.5_{\pm 5.0}$	$22.4_{\pm 1.7}$	$47.4_{\pm 2.0}$	$30.1_{\pm 6.9}$	$19.7_{\pm 1.5}$	$57.0_{\pm 2.3}$	$43.6_{\pm 1.0}$
DyHead O365	5	Full	$43.5_{\pm 1.0}$	$25.3_{\pm 1.8}$	$35.8_{\pm 0.5}$	$63.0_{\pm 1.0}$	$56.2_{\pm 3.9}$	$76.8_{\pm 5.9}$	$62.5_{\pm 8.7}$	$46.6_{\pm 3.1}$	$28.8_{\pm 2.2}$	$51.2_{\pm 2.2}$	$38.7_{\pm 4.1}$	$21.0_{\pm 1.4}$	$53.4_{\pm 5.2}$	$46.4_{\pm 1.1}$
DyHead O365	10	Full	$46.6_{\pm 0.3}$	$29.0_{\pm 2.8}$	$41.7_{\pm 1.0}$	$65.2_{\pm 2.5}$	$62.5_{\pm 0.8}$	$85.4_{\pm 2.2}$	$67.9_{\pm 4.5}$	$47.9_{\pm 2.2}$	$28.6_{\pm 5.0}$	$53.8_{\pm 1.0}$	$39.2_{\pm 4.9}$	$27.9_{\pm 2.3}$	$64.1_{\pm 2.6}$	$50.8_{\pm 1.3}$
DyHead O365	All	Full	53.3	28.4	49.5	73.5	77.9	84.0	69.2	56.2	43.6	59.2	68.9	53.7	73.7	60.8
GLIP-T	1	Prompt	$54.4_{\pm 0.9}$	$15.2_{\pm 1.4}$	$32.5_{\pm 1.0}$	$68.0_{\pm 3.2}$	$60.0_{\pm 0.7}$	$75.8_{\pm 1.2}$	$72.3_{\pm 0.0}$	$54.5_{\pm 3.9}$	$24.1_{\pm 3.0}$	$59.2_{\pm 0.9}$	$57.4_{\pm 0.6}$	$18.9_{\pm 1.8}$	$56.9_{\pm 2.7}$	$49.9_{\pm 0.6}$
GLIP-T	3	Prompt	$56.8_{\pm 0.8}$	$18.9_{\pm 3.6}$	$37.6_{\pm 1.6}$	$72.4_{\pm 0.5}$	$62.8_{\pm 1.3}$	$85.4_{\pm 2.8}$	$64.5_{\pm 4.6}$	$69.1_{\pm 1.8}$	$22.0_{\pm 0.9}$	$62.7_{\pm 1.1}$	$56.1_{\pm 0.6}$	$25.9_{\pm 0.7}$	$63.8_{\pm 4.8}$	$53.7_{\pm 1.3}$
GLIP-T	5	Prompt	$58.5_{\pm 0.5}$	$18.2_{\pm 0.1}$	$41.0_{\pm 1.2}$	$71.8_{\pm 2.4}$	$65.7_{\pm 0.7}$	$87.5_{\pm 2.2}$	$72.3_{\pm 0.0}$	$60.6_{\pm 2.2}$	$31.4_{\pm 4.2}$	$61.0_{\pm 1.8}$	$54.4_{\pm 0.6}$	$32.6_{\pm 1.4}$	$66.3_{\pm 2.8}$	$55.5_{\pm 0.5}$
GLIP-T	10	Prompt	$59.7_{\pm 0.7}$	$19.8_{\pm 1.6}$	$44.8_{\pm 0.9}$	$72.1_{\pm 2.0}$	$65.9_{\pm 0.6}$	$87.4_{\pm 1.1}$	$72.3_{\pm 0.0}$	$57.5_{\pm 1.2}$	$30.0_{\pm 1.4}$	$62.1_{\pm 1.4}$	$57.8_{\pm 0.9}$	$33.5_{\pm 0.1}$	$73.1_{\pm 1.4}$	$56.6_{\pm 0.2}$
GLIP-T	All	Prompt	66.4	27.6	50.9	70.6	73.3	88.1	67.7	64.0	40.3	65.4	68.3	50.7	78.5	62.4
GLIP-T	1	Full	$54.8_{\pm 2.0}$	$18.4_{\pm 1.0}$	$33.8_{\pm 1.1}$	$70.1_{\pm 2.9}$	$64.2_{\pm 1.8}$	$83.7_{\pm 3.0}$	$70.8_{\pm 2.1}$	$56.2_{\pm 1.8}$	$22.9_{\pm 0.2}$	$56.6_{\pm 0.5}$	$59.9_{\pm 0.4}$	$18.9_{\pm 1.3}$	$54.5_{\pm 2.7}$	$51.1_{\pm 0.1}$
GLIP-T	3	Full	$58.1_{\pm 0.5}$	$22.9_{\pm 1.3}$	$40.8_{\pm 0.9}$	$65.7_{\pm 1.6}$	$66.0_{\pm 0.2}$	$84.7_{\pm 0.5}$	$65.7_{\pm 2.8}$	$62.6_{\pm 1.4}$	$27.2_{\pm 2.7}$	$61.9_{\pm 1.8}$	$60.7_{\pm 0.2}$	$27.1_{\pm 1.2}$	$70.4_{\pm 2.5}$	$54.9_{\pm 0.2}$
GLIP-T	5	Full	$59.5_{\pm 0.4}$	$23.8_{\pm 0.9}$	$43.6_{\pm 1.4}$	$68.7_{\pm 1.3}$	$66.1_{\pm 0.6}$	$85.4_{\pm 0.4}$	$72.3_{\pm 0.0}$	$62.1_{\pm 2.0}$	$27.3_{\pm 1.2}$	$61.0_{\pm 1.8}$	$62.7_{\pm 1.6}$	$34.5_{\pm 0.5}$	$66.6_{\pm 2.3}$	$56.4_{\pm 0.4}$
GLIP-T	10	Full	$59.1_{\pm 1.3}$	$26.3_{\pm 1.1}$	$46.3_{\pm 1.6}$	$67.3_{\pm 1.5}$	$67.1_{\pm 0.7}$	$87.8_{\pm 0.5}$	$72.3_{\pm 0.0}$	$57.7_{\pm 1.7}$	$34.6_{\pm 1.7}$	$65.4_{\pm 1.4}$	$61.6_{\pm 1.0}$	$39.3_{\pm 1.0}$	$74.7_{\pm 2.3}$	$58.4_{\pm 0.2}$
GLIP-T	All	Full	62.3	31.2	52.5	70.8	78.7	88.1	75.6	61.4	51.4	65.3	71.2	58.7	76.7	64.9
GLIP-L	1	Prompt	$62.8_{\pm 0.4}$	$18.0_{\pm 1.8}$	$37.4_{\pm 0.3}$	$71.9_{\pm 2.4}$	$68.9_{\pm 0.1}$	$81.8_{\pm 3.4}$	$65.0_{\pm 2.8}$	$63.9_{\pm 0.4}$	$70.2_{\pm 1.2}$	$67.0_{\pm 0.4}$	$69.3_{\pm 0.1}$	$27.6_{\pm 0.4}$	$69.8_{\pm 0.6}$	$59.5_{\pm 0.4}$
GLIP-L	3	Prompt	$65.0_{\pm 0.5}$	$21.4_{\pm 1.0}$	$43.6_{\pm 1.1}$	$72.9_{\pm 0.7}$	$70.4_{\pm 0.1}$	$91.4_{\pm 0.7}$	$57.7_{\pm 3.7}$	$70.7_{\pm 1.1}$	$69.7_{\pm 0.9}$	$62.6_{\pm 0.8}$	$67.7_{\pm 0.4}$	$36.2_{\pm 1.1}$	$68.8_{\pm 1.5}$	$61.4_{\pm 0.3}$
GLIP-L	5	Prompt	$65.6_{\pm 0.3}$	$19.9_{\pm 1.6}$	$47.7_{\pm 0.7}$	$73.7_{\pm 0.7}$	$70.6_{\pm 0.3}$	$86.8_{\pm 0.5}$	$64.6_{\pm 0.7}$	$69.4_{\pm 3.3}$	$68.0_{\pm 1.3}$	$67.8_{\pm 1.5}$	$68.3_{\pm 0.3}$	$36.6_{\pm 1.6}$	$71.9_{\pm 0.6}$	$62.4_{\pm 0.5}$
GLIP-L	10	Prompt	$65.9_{\pm 0.2}$	$23.4_{\pm 2.6}$	$50.3_{\pm 0.7}$	$73.6_{\pm 0.7}$	$71.8_{\pm 0.3}$	$86.5_{\pm 0.3}$	$70.5_{\pm 1.1}$	$69.0_{\pm 0.5}$	$69.4_{\pm 2.4}$	$70.8_{\pm 1.2}$	$68.8_{\pm 0.6}$	$39.3_{\pm 0.9}$	$74.9_{\pm 2.1}$	$64.2_{\pm 0.4}$
GLIP-L	All	Prompt	72.9	23.0	51.8	72.0	75.8	88.1	75.2	69.5	73.6	72.1	73.7	53.5	81.4	$67.9_{\pm 0.0}$
GLIP-L	1	Full	$64.8_{\pm 0.6}$	$18.7_{\pm 0.6}$	$39.5_{\pm 1.2}$	$70.0_{\pm 1.5}$	$70.5_{\pm 0.2}$	$69.8_{\pm 18.0}$	$70.6_{\pm 4.0}$	$68.4_{\pm 1.2}$	$71.0_{\pm 1.3}$	$65.4_{\pm 1.1}$	$68.1_{\pm 0.2}$	$28.9_{\pm 2.9}$	$72.9_{\pm 4.7}$	$59.9_{\pm 1.4}$
GLIP-L	3	Full	$65.6_{\pm 0.6}$	$22.3_{\pm 1.1}$	$45.2_{\pm 0.4}$	$72.3_{\pm 1.4}$	$70.4_{\pm 0.4}$	$81.6_{\pm 13.3}$	$71.8_{\pm 0.3}$	$65.3_{\pm 1.6}$	$67.6_{\pm 1.0}$	$66.7_{\pm 0.9}$	$68.1_{\pm 0.3}$	$37.0_{\pm 1.9}$	$73.1_{\pm 3.3}$	$62.1_{\pm 0.7}$
GLIP-L	5	Full	$66.6_{\pm 0.4}$	$26.4_{\pm 2.5}$	$49.5_{\pm 1.1}$	$70.7_{\pm 0.2}$	$71.9_{\pm 0.2}$	$88.1_{\pm 0.0}$	$71.1_{\pm 0.6}$	$68.8_{\pm 1.2}$	$68.5_{\pm 1.7}$	$70.0_{\pm 0.9}$	$68.3_{\pm 0.5}$	$39.9_{\pm 1.4}$	$75.2_{\pm 2.7}$	$64.2_{\pm 0.3}$
GLIP-L	10	Full	$66.4_{\pm 0.7}$	$32.0_{\pm 1.4}$	$52.3_{\pm 1.1}$	$70.6_{\pm 0.7}$	$72.4_{\pm 0.3}$	$88.1_{\pm 0.0}$	$67.1_{\pm 3.6}$	$64.7_{\pm 3.1}$	$69.4_{\pm 1.4}$	$71.5_{\pm 0.8}$	$68.4_{\pm 0.7}$	$44.3_{\pm 0.6}$	$76.3_{\pm 1.1}$	$64.9_{\pm 0.7}$
GLIP-L	All	Full	69.6	32.6	56.6	76.4	79.4	88.1	67.1	69.4	65.8	71.6	75.7	60.3	83.1	68.9
GLIPv2-T	1	Prompt	$51.2_{\pm 0.3}$	$17.7_{\pm 1.2}$	$34.2_{\pm 0.1}$	$68.7_{\pm 1.2}$	$67.3_{\pm 0.9}$	$83.7_{\pm 2.1}$	$68.1_{\pm 1.7}$	$53.4_{\pm 0.2}$	$30.0_{\pm 0.9}$	$59.0_{\pm 0.1}$	$60.0_{\pm 0.3}$	$21.9_{\pm 0.2}$	$66.5_{\pm 0.7}$	$52.4_{\pm 0.5}$
GLIPv2-T	3	Prompt	$66.6_{\pm 0.2}$	$11.5_{\pm 0.7}$	$37.2_{\pm 1.0}$	$71.7_{\pm 0.3}$	$70.1_{\pm 0.4}$	$45.7_{\pm 0.1}$	$57.7_{\pm 1.2}$	$69.7_{\pm 1.5}$	$42.7_{\pm 0.4}$	$67.5_{\pm 0.9}$	$65.6_{\pm 1.0}$	$36.7_{\pm 1.2}$	$69.2_{\pm 1.2}$	$55.6_{\pm 0.4}$
GLIPv2-T	5	Prompt	$58.9_{\pm 1.2}$	$17.4_{\pm 0.6}$	$42.8_{\pm 0.4}$	$72.6_{\pm 0.5}$	$66.1_{\pm 0.2}$	$84.9_{\pm 0.8}$	$69.7_{\pm 0.6}$	$65.5_{\pm 2.1}$	$35.6_{\pm 0.8}$	$62.8_{\pm 0.9}$	$59.8_{\pm 0.2}$	$35.5_{\pm 0.9}$	$74.4_{\pm 0.2}$	$57.4_{\pm 0.4}$
GLIPv2-T	10	Prompt	$59.9_{\pm 0.4}$	$21.6_{\pm 2.0}$	$43.7_{\pm 0.3}$	$74.3_{\pm 0.4}$	$68.2_{\pm 0.7}$	$88.1_{\pm 0.1}$	$72.0_{\pm 0.9}$	$60.0_{\pm 0.4}$	$35.6_{\pm 1.2}$	$66.1_{\pm 0.6}$	$61.0_{\pm 0.3}$	$42.8_{\pm 0.4}$	$70.9_{\pm 3.2}$	$58.8_{\pm 0.5}$
GLIPv2-T	All	Prompt	67.4	22.3	50.5	74.3	73.4	85.5	74.7	65.8	53.7	67.4	68.9	52.3	83.7	$64.8_{\pm 0.0}$
GLIPv2-T	1	Full	$64.8_{\pm 0.6}$	$18.7_{\pm 0.6}$	$39.5_{\pm 1.2}$	$70.0_{\pm 1.5}$	$70.5_{\pm 0.2}$	$69.8_{\pm 18.0}$	$70.6_{\pm 4.0}$	$68.4_{\pm 1.2}$	$71.0_{\pm 1.3}$	$65.4_{\pm 1.1}$	$68.1_{\pm 0.2}$	$28.9_{\pm 2.9}$	$72.9_{\pm 4.7}$	$52.8_{\pm 1.4}$
GLIPv2-T	3	Full	$53.9_{\pm 0.1}$	$17.8_{\pm 0.7}$	$42.7_{\pm 1.1}$	$73.1_{\pm 1.0}$	$65.9_{\pm 0.2}$	$84.7_{\pm 3.4}$	$69.7_{\pm 0.8}$	$60.7_{\pm 1.3}$	$28.8_{\pm 0.8}$	$61.7_{\pm 1.3}$	$60.6_{\pm 0.2}$	$35.5_{\pm 0.4}$	$68.3_{\pm 1.7}$	$55.6_{\pm 0.7}$
GLIPv2-T	5	Full	$58.9_{\pm 0.2}$	$17.4_{\pm 1.1}$	$42.8_{\pm 1.3}$	$72.6_{\pm 0.7}$	$66.1_{\pm 0.6}$	$84.9_{\pm 0.9}$	$69.7_{\pm 0.3}$	$65.5_{\pm 1.0}$	$35.6_{\pm 0.9}$	$62.8_{\pm 0.3}$	$59.8_{\pm 0.2}$	$35.5_{\pm 1.2}$	$74.4_{\pm 2.1}$	$57.4_{\pm 0.4}$
GLIPv2-T	10	Full	$57.6_{\pm 1.0}$	$27.6_{\pm 1.2}$	$49.1_{\pm 1.0}$	$70.4_{\pm 0.5}$	$69.2_{\pm 0.2}$	$88.1_{\pm 0.0}$	$73.1_{\pm 2.3}$	$58.0_{\pm 2.8}$	$42.9_{\pm 1.2}$	$64.8_{\pm 0.2}$	$62.1_{\pm 0.9}$	$39.9_{\pm 0.4}$	$71.6_{\pm 0.8}$	$59.7_{\pm 0.3}$
GLIPv2-T	All	Full	66.4	30.2	52.5	74.8	80.0	88.1	74.3	63.7	54.4	63.0	73.0	60.1	83.5	66.5
GLIPv2-B	1	Prompt	$68.7_{\pm 0.1}$	$19.9_{\pm 0.3}$	$38.4_{\pm 0.8}$	$68.5_{\pm 1.0}$	$68.6_{\pm 0.8}$	$87.7_{\pm 3.0}$	$69.3_{\pm 1.7}$	$68.5_{\pm 0.4}$	$55.2_{\pm 0.3}$	$65.7_{\pm 0.7}$	$67.2_{\pm 0.1}$	$34.8_{\pm 0.8}$	$69.6_{\pm 0.4}$	$60.4_{\pm 0.3}$
GLIPv2-B	3	Prompt	$67.2_{\pm 0.6}$	$22.2_{\pm 0.3}$	$46.5_{\pm 0.9}$	$71.2_{\pm 0.8}$	$70.9_{\pm 0.1}$	$86.9_{\pm 0.2}$	$67.7_{\pm 1.8}$	$63.7_{\pm 2.3}$	$46.9_{\pm 0.8}$	$68.1_{\pm 0.4}$	$67.4_{\pm 0.9}$	$47.9_{\pm 1.0}$	$78.9_{\pm 1.7}$	$62.0_{\pm 0.5}$
GLIPv2-B	5	Prompt	$68.9_{\pm 1.0}$	$25.7_{\pm 0.4}$	$50.5_{\pm 0.9}$	$73.8_{\pm 1.5}$	$69.7_{\pm 0.6}$	$84.9_{\pm 0.3}$	$69.3_{\pm 0.7}$	$65.8_{\pm 1.6}$	$65.7_{\pm 1.0}$	$69.2_{\pm 0.3}$	$67.5_{\pm 0.7}$	$34.0_{\pm 0.2}$	$73.1_{\pm 0.6}$	$62.9_{\pm 0.4}$
GLIPv2-B	10	Prompt	$69.4_{\pm 0.7}$	$21.8_{\pm 1.3}$	$48.7_{\pm 0.2}$	$71.3_{\pm 0.2}$	$71.0_{\pm 0.7}$	$88.1_{\pm 0.4}$	$68.6_{\pm 0.7}$	$73.5_{\pm 0.3}$	$61.5_{\pm 1.9}$	$69.3_{\pm 0.2}$	$68.6_{\pm 0.7}$	$41.3_{\pm 0.2}$	$75.2_{\pm 1.3}$	$63.8_{\pm 0.3}$
GLIPv2-B	All	Prompt	71.9	26.1	50.6	74.5	73.5	86.9	74.9	71.0	71.6	71.0	72.4	50.2	80.5	$67.3_{\pm 0.0}$
GLIPv2-B	1	Full	$67.8_{\pm 0.4}$	$18.7_{\pm 0.3}$	$44.2_{\pm 0.9}$	$71.4_{\pm 0.3}$	$70.4_{\pm 1.2}$	$87.9_{\pm 7.3}$	$66.1_{\pm 2.4}$	$68.9_{\pm 1.1}$	$60.6_{\pm 1.6}$	$68.1_{\pm 0.6}$	$69.0_{\pm 0.7}$	$35.1_{\pm 0.9}$	$68.9_{\pm 2.1}$	$61.2_{\pm 0.6}$
GLIPv2-B	3	Full	$68.1_{\pm 0.2}$	$25.7_{\pm 0.4}$	$46.4_{\pm 1.6}$	$69.8_{\pm 1.3}$	$71.3_{\pm 1.2}$	$88.0_{\pm 3.4}$	$68.6_{\pm 0.9}$	$69.8_{\pm 1.7}$	$60.1_{\pm 0.3}$	$68.4_{\pm 1.9}$	$68.5_{\pm 0.6}$	$39.8_{\pm 0.8}$	$71.4_{\pm 2.1}$	$62.8_{\pm 0.8}$
GLIPv2-B	5	Full	$68.6_{\pm 1.0}$	$21.6_{\pm 0.6}$	$46.7_{\pm 0.7}$	$70.9_{\pm 0.9}$	$71.0_{\pm 1.2}$	$88.1_{\pm 3.7}$	$69.1_{\pm 0.2}$	$71.8_{\pm 1.0}$	$61.5_{\pm 0.7}$	$68.7_{\pm 0.2}$	$69.3_{\pm 0.8}$	$40.2_{\pm 1.0}$	$74.8_{\pm 2.8}$	$63.3_{\pm 0.6}$
GLIPv2-B	10	Full	$67.4_{\pm 1.3}$	$22.3_{\pm 1.1}$	$50.5_{\pm 0.7}$	$74.3_{\pm 0.4}$	$73.4_{\pm 0.4}$	$85.5_{\pm 0.1}$	$74.7_{\pm 0.9}$	$65.8_{\pm 2.4}$	$53.7_{\pm 1.1}$	$67.4_{\pm 0.9}$	$68.9_{\pm 0.7}$	$52.3_{\pm 0.6}$	$83.7_{\pm 3.2}$	$64.6_{\pm 0.3}$
GLIPv2-B	All	Full	71.1	32.6	57.5	73.6	80.0	88.1	74.9	68.2	70.6	71.2	76.5	58.7	79.6	69.4
GLIPv2-H	1	Prompt	$68.3_{\pm 0.6}$	$16.4_{\pm 0.6}$	$45.8_{\pm 0.3}$	$72.0_{\pm 0.5}$	$67.9_{\pm 0.9}$	$89.3_{\pm 3.2}$	$69.3_{\pm 1.7}$	$67.9_{\pm 0.8}$	$66.3_{\pm 1.9}$	$68.0_{\pm 0.7}$	$66.8_{\pm 0.3}$	$33.9_{\pm 0.4}$	$70.7_{\pm 1.5}$	$61.4_{\pm 0.5}$
GLIPv2-H	3	Prompt	$69.5_{\pm 0.7}$	$25.9_{\pm 0.2}$	$50.0_{\pm 1.2}$	$75.4_{\pm 1.4}$	$70.1_{\pm 0.9}$	$85.9_{\pm 2.5}$	$69.3_{\pm 0.7}$	$70.8_{\pm 1.2}$	$66.4_{\pm 0.8}$	$68.0_{\pm 1.2}$	$68.8_{\pm 0.9}$	$34.0_{\pm 0.3}$	$72.7_{\pm 1.6}$	$63.6_{\pm 0.6}$
GLIPv2-H	5	Prompt	$69.4_{\pm 0.7}$	$22.0_{\pm 0.6}$	$49.1_{\pm 0.1}$	$70.7_{\pm 1.0}$	$73.0_{\pm 0.5}$	$88.1_{\pm 0.8}$	$70.3_{\pm 0.4}$	$71.2_{\pm 1.8}$	$62.9_{\pm 1.4}$	$70.1_{\pm 0.3}$	$68.3_{\pm 0.6}$	$42.7_{\pm 0.6}$	$74.3_{\pm 0.5}$	$63.9_{\pm 0.7}$
GLIPv2-H	10	Prompt	$66.0_{\pm 0.7}$	$27.5_{\pm 1.3}$	$53.8_{\pm 0.2}$	$74.6_{\pm 0.2}$	$80.1_{\pm 0.7}$	$87.4_{\pm 0.4}$	$69.3_{\pm 0.7}$	$66.0_{\pm 0.3}$	$51.2_{\pm 1.9}$	$67.2_{\pm 0.2}$	$72.8_{\pm 0.7}$	$58.3_{\pm 0.2}$	$76.5_{\pm 1.3}$	$65.5_{\pm 0.6}$
GLIPv2-H	All	Prompt	71.2	31.1	57.1	75.0	79.8	88.1	68.6	68.3	59.6	70.9	73.6	61.4	78.6	$69.1_{\pm 0.0}$
GLIPv2-H	1	Full	$67.8_{\pm 0.6}$	$17.3_{\pm 0.6}$	$50.7_{\pm 0.3}$	$63.8_{\pm 0.5}$	$67.3_{\pm 0.9}$	$89.4_{\pm 3.2}$	$69.3_{\pm 1.7}$	$68.2_{\pm 0.8}$	$66.6_{\pm 1.9}$	$66.8_{\pm 0.7}$	$67.0_{\pm 0.3}$	$34.0_{\pm 0.4}$	$75.0_{\pm 1.5}$	$61.7_{\pm 0.5}$
GLIPv2-H	3	Full	$62.3_{\pm 0.2}$	$29.1_{\pm 0.4}$	$53.8_{\pm 1.6}$	$72.7_{\pm 1.3}$	$78.4_{\pm 1.2}$	$85.8_{\pm 3.4}$	$68.6_{\pm 0.9}$	$60.7_{\pm 1.7}$	$43.6_{\pm 0.3}$	$65.9_{\pm 1.9}$	$72.2_{\pm 0.6}$	$55.9_{\pm 0.8}$	$81.1_{\pm 2.1}$	$64.1_{\pm 0.8}$
GLIPv2-H	5	Full	$66.4_{\pm 1.0}$	$23.4_{\pm 0.6}$	$50.7_{\pm 0.7}$	$73.9_{\pm 0.9}$	$71.8_{\pm 1.2}$	$84.2_{\pm 3.7}$	$71.2_{\pm 0.2}$	$68.1_{\pm 1.0}$	$67.4_{\pm 0.7}$	$70.8_{\pm 0.2}$	$65.8_{\pm 0.8}$	$54.6_{\pm 1.0}$	$75.6_{\pm 2.8}$	$64.4_{\pm 0.6}$
GLIPv2-H	10	Full	$67.3_{\pm 1.3}$	$31.6_{\pm 1.1}$	$52.4_{\pm 0.7}$	$71.3_{\pm 0.4}$	$80.0_{\pm 0.4}$	$88.1_{\pm 0.1}$	$72.9_{\pm 0.9}$	$56.9_{\pm 2.4}$	$52.2_{\pm 1.1}$	$65.4_{\pm 0.9}$	$73.9_{\pm 0.7}$	$61.0_{\pm 0.6}$	$84.0_{\pm 3.2}$	$65.9_{\pm 0.3}$
GLIPv2-H	All	Full	74.4	36.3	58.7	77.1	79.3	88.1	74.3	73.1	70.0	72.2	72.5	58.3	81.4	70.4

Table 12: Per-dataset performance of DyHead, GLIP-T, GLIP-L, and GLIPv2-T, GLIPv2-B and GLIPv2-H. For PascalVOC, we report the mAP (IoU=0.50:0.95) using the COCO evaluation script, to be consistent with other 12 datasets. "Prompt" denotes prompt tuning. "Full" denotes full-model tuning.