ReCLIP: A Strong Zero-Shot Baseline for Referring Expression Comprehension

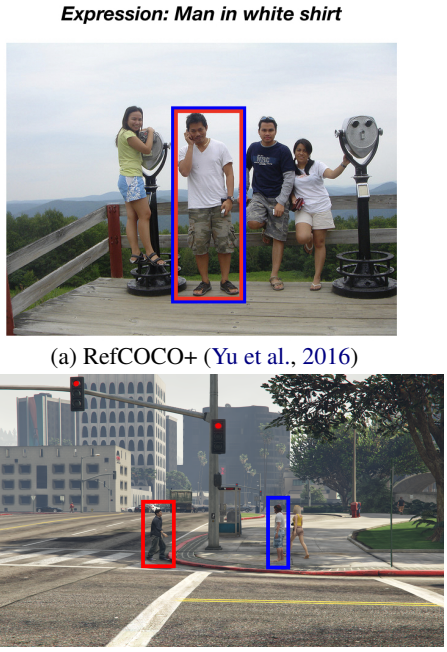
Anonymous ACL submission

Abstract

Training a referring expression comprehension (ReC) model for a new visual domain requires collecting referring expressions, and potentially corresponding bounding boxes, for images in the domain. While large-scale pre-trained models are useful for image classification across domains, it remains unclear if they can be applied in a zero-shot manner to more complex tasks like ReC. We present ReCLIP, a simple but strong *zero-shot* baseline that repurposes CLIP, a state-of-the-art large-scale model, for ReC. Motivated by the close connection between ReC and CLIP's contrastive pre-training objective, the first component of ReCLIP is a region-scoring method that isolates object proposals via cropping and blurring, and passes them to CLIP. However, through controlled experiments on a synthetic dataset, we find that CLIP is largely incapable of performing spatial reasoning off-the-shelf. Thus, the second component of ReCLIP is a spatial relation resolver that handles several types of spatial relations. We reduce the gap between zero-shot baselines from prior work and supervised models by as much as 30% on RefCOCOg, and on RefGTA (video game imagery), we outperform supervised ReC models trained on real images by an absolute 12%.

1 Introduction

Visual referring expression comprehension (ReC)—the task of localizing an object in an image given a textual referring expression—has applications in a broad range of visual domains. For example, ReC is useful for guiding a robot in the real world (Shridhar et al., 2020) and also for creating natural language interfaces for software applications with visuals (Wichers et al., 2018). Though the task is the same across domains, the domain shift is problematic for supervised referring expression models, as shown in Figure 1: the same simple referring expression is localized correctly in the training domain but incorrectly in a new domain.



(a) RefCOCO+ (Yu et al., 2016)

(b) RefGTA (Tanaka et al., 2019)

Figure 1: Predictions from ReCLIP (blue) and UNITER-Large (Chen et al., 2020) (red) for the same referring expression on images from two visual domains. UNITER-Large fails on the GTA (video game) domain, while ReCLIP selects the correct proposal in both cases.

Collecting task-specific data in each domain of interest is expensive. Weakly supervised ReC (Rohrbach et al., 2016) partially addresses this issue, since it does not require the ground-truth box for each referring expression, but it still assumes the availability of referring expressions paired with images and trains on these. Given a large-scale pre-trained vision and language model and a method for doing ReC zero-shot—i.e. without any additional training—practitioners could save a great deal of time and effort. Moreover, as pre-trained models have become more accurate via scaling (Kaplan et al., 2020), fine-tuning the best models has become prohibitively expensive–and sometimes infeasible because the model is offered only via API, e.g. GPT-3 (Brown et al., 2020).

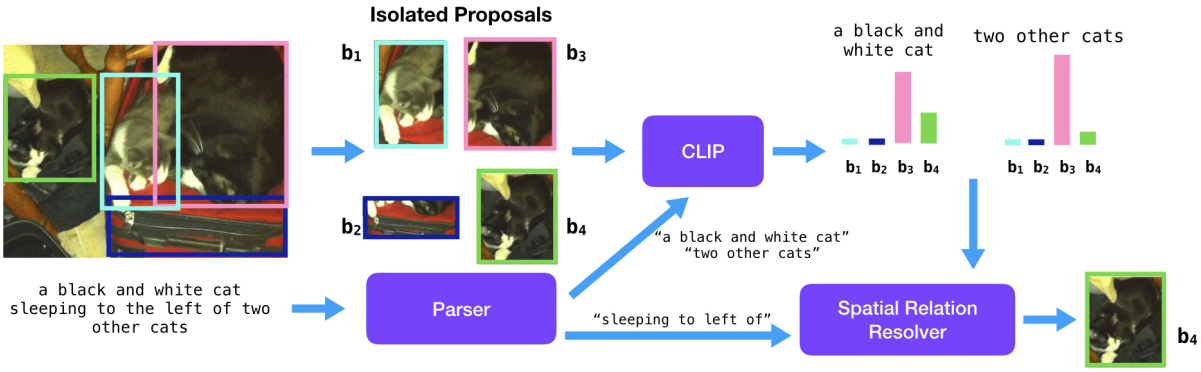


Figure 2: Overview of ReCLIP. Given object proposals, we isolate the corresponding image regions by cropping and blurring (only cropping shown here). Using a parser, we extract the noun chunks of the expression. For each noun chunk, CLIP outputs a distribution over proposals. The relations from the parser and CLIP's probabilities are combined by a spatial relation resolver to select the final proposal. In this example, CLIP ranks b_3 highest for both noun chunks, but using the relation resolver we obtain the correct answer b_4.

Pre-trained vision and language models like CLIP (Radford et al., 2021) achieve strong zero-shot performance in image classification across visual domains (Jia et al., 2021) and in object detection (Gu et al., 2021), but the same success has not yet been achieved in tasks requiring reasoning over vision and language. For example, Shen et al. (2021) show that a straightforward zero-shot approach for VQA using CLIP performs poorly. Specific to ReC, Yao et al. (2021) introduce a zero-shot approach via Colorful Prompt Tuning (CPT), which colors object proposals and references the color in the text prompt to score proposals, but this has low accuracy. In both of these cases, the proposed zero-shot method is not aligned closely enough with the model's pre-training task of matching naturally occurring images and captions.

In this work, we propose ReCLIP, a simple but strong new baseline for zero-shot ReC. ReCLIP, illustrated in Figure 2, has two key components: a method for scoring object proposals using CLIP and a method for handling spatial relations between objects. Our method for scoring region proposals, Isolated Proposal Scoring (IPS), effectively reduces ReC to the contrastive pre-training task used by CLIP and other models. Specifically, we propose to isolate individual proposals via cropping and blurring the images and to score these isolated proposals with the given expression using CLIP.

To handle relations between objects, we first consider whether CLIP encodes the spatial information necessary to resolve these relations. We show through a controlled experiment on CLEVR images (Johnson et al., 2017) that CLIP and another pre-trained model ALBEF (Li et al., 2021) are unable to perform its pre-training task on examples that require spatial reasoning. Thus, any method that solely relies on these models is unlikely to resolve spatial relations accurately. Consequently, we propose spatial heuristics for handling spatial relations in which an expression is decomposed into subqueries, CLIP is used to compute proposal probabilities for each subquery, and the outputs for all subqueries are combined with simple rules.

On the standard RefCOCO/g/+ datasets (Mao et al., 2016; Yu et al., 2016), we find that ReCLIP outperforms CPT (Yao et al., 2021) by more than 20%. Compared to a stronger GradCAM (Selvaraju et al., 2019) baseline, ReCLIP obtains better accuracy on average and has less variance across object types. Finally, in order to illustrate the practical value of zero-shot grounding, we also demonstrate that our zero-shot method surpasses the out-of-domain performance of state-of-the-art supervised ReC models. We evaluate on the RefGTA dataset (Tanaka et al., 2019), which contains images from a video game (out of domain for models trained only on real photos). Using ReCLIP and an object detector trained outside the target domain, we outperform UNITER-Large (Chen et al., 2020) (using the same proposals) and MDETR (Kamath et al., 2021) by an absolute 12%.

In summary, our contributions include: (1) ReCLIP, a zero-shot method for referring expression comprehension, (2) showing that CLIP has low zero-shot spatial reasoning performance, and (3) a comparison of our zero-shot ReC performance with the out-of-domain performance of state-of-the-art fully supervised ReC systems.[1]

[1]Our code will be released upon publication.

2 Background

In this section, we first describe the task at hand (§2.1) and introduce CLIP, the pre-trained model we primarily use (§2.2). We then describe two existing methods for scoring region proposals using a pre-trained vision and language model: colorful prompt tuning (§2.3) and GradCAM (§2.4).

2.1 Task description

In referring expression comprehension (ReC), the model is given an image and a textual referring expression describing an entity in the image. The goal of the task is to select the object (bounding box) that best matches the expression. As in much of the prior work on REC, we assume access to a set of object proposals $b_1, b_2, ..., b_n$, each of which is a bounding box in the image. Task accuracy is measured as the percentage of instances for which the model selects a proposal whose intersection-over-union (IoU) with the ground-truth box is at least 0.5. In this paper, we focus on the *zero-shot* setting in which we apply a pre-trained model to ReC without using any training data for the task.

2.2 Pre-trained model architecture

The zero-shot approaches that we consider are general in that the only requirement for the pre-trained model is that when given a *query* consisting of an image and text, it computes a score for the similarity between the image and text. In this paper, we primarily use CLIP (Radford et al., 2021). We focus on CLIP because it was pre-trained on 400M image-caption pairs collected from the web[2] and therefore achieves impressive zero-shot image classification performance on a variety of visual domains. CLIP has an image-only encoder, which is either a ResNet-based architecture (He et al., 2016) or a visual transformer (Dosovitskiy et al., 2021), and a text-only transformer. We mainly use the RN50x16 and ViT-B/32 versions of CLIP. The image encoder takes the raw image and produces an image representation $\mathbf{x} \in \mathbb{R}^d$, and the text transformer takes the sequence of text tokens and produces a text representation $\mathbf{y} \in \mathbb{R}^d$. In CLIP's contrastive pre-training task, given a batch of N images and matching captions, each image must be matched with the corresponding text. The model's probability of matching image i with caption j is given by $\exp(\beta \mathbf{x_i}^T \mathbf{y_j}) / \sum_{k=1}^{N} \exp(\beta \mathbf{x_i}^T \mathbf{y_k})$, where β is a hyperparameter.[3]

We now describe two techniques from prior work for selecting a proposal using a pre-trained model.

2.3 Colorful Prompt Tuning (CPT)

The first baseline from prior work that we consider is colorful prompt tuning (CPT), proposed by Yao et al. (2021) [4]: they shade proposals with different colors and use a masked language prompt in which the referring expression is followed by "in [MASK] color". The color with the highest probability from a pre-trained masked language model (MLM) (VinVL; (Zhang et al., 2021)) is then chosen. In order to apply this method to models like CLIP, that provide image-text scores but do not offer an MLM, we create a version of the input image for each proposal, where the proposal is transparently shaded in red.[5] Our template for the input text is "[referring expression] is in red color." Since we have adapted CPT for non-MLM models, we refer to this method as *CPT-adapted* in the experiments.

2.4 Gradient-based visualizations

The second baseline from prior work that we consider is based on gradient-based visualizations, which are a popular family of techniques for understanding, on a range of computer vision tasks, which part(s) of an input image are most important to a model's prediction. We focus on the most popular technique in this family, GradCAM (Selvaraju et al., 2019). Our usage of GradCAM follows Li et al. (2021), in which GradCAM is used to perform weakly supervised referring expression comprehension using the ALBEF model. In our setting, for a given layer in a visual transformer, we take the layer's class-token (CLS) attention matrix $M \in \mathbb{R}^{h,w}$. The spatial dimensions h and w are dependent on the model's architecture and are generally smaller than the input dimensions of the image. Then the GradCAM is computed as $G = M \odot \frac{\partial L}{\partial M}$, where L is the model's output logit (the similarity score for the image-text pair) and $\odot$ denotes elementwise multiplication. The procedure for applying GradCAM when the visual encoder is a convolutional network is similar; in

[2] This dataset is not public.

[3] $\mathbf{x_i}$ and $\mathbf{y_i}$ are normalized before the dot product.

[4] CPT is the name given by Yao et al. (2021), but note that we do not perform few-shot/supervised tuning.

[5] Specifically, we use the RGB values (240, 0, 30) and transparency 127/255 that Yao et al. (2021) say works best with their method. An example is shown in Appendix B.

place of the attention matrix, we use the activations of the final convolutional layer. Next, we perform a bicubic interpolation on G so that it has the same dimensions as the input image. Finally, we compute for each proposal $b_i = (x_1, y_1, x_2, y_2)$ the score $\frac{1}{A^\alpha} \sum_{i=x_1}^{x_2} \sum_{j=y_1}^{y_2} G[i, j]$, where A is the area of the image and α is a hyperparameter, and we choose the proposal with the highest score.

3 ReCLIP

ReCLIP consists of two main components: (1) a region-scoring method that is different from CPT and GradCAM and (2) a rule-based relation resolver. In this section, we first describe our region scoring method (§3.1). However, using controlled experiments on a synthetic dataset, we find that CLIP has poor zero-shot spatial reasoning performance (§3.2). Therefore, we propose a system that uses heuristics to resolve spatial relations (§3.3).

3.1 Isolated Proposal Scoring (IPS)

Our proposed method, which we call *isolated proposal scoring*, is based on the observation that ReC is similar to the contrastive learning task with which models like CLIP are pre-trained, except that rather than selecting one out of several images to match with a given text, we must select one out of several image regions. Therefore, for each proposal, we create a new image in which that proposal is isolated. We consider two methods of isolation – *cropping* the image to contain only the proposal and *blurring* everything in the image except for the proposal region. For blurring, we apply a Gaussian filter with standard deviation σ to the image RGB values. Appendix A.2 provides an example of isolation by blurring. The score for an isolated proposal is obtained by passing it and the expression through the pre-trained model. To use cropping and blurring in tandem, we obtain a score s_{crop} and s_{blur} for each proposal and use $s_{crop} + s_{blur}$ as the final score. This can be viewed as an ensemble of "visual prompts," analogous to Radford et al. (2021)'s ensembling of text prompts.

3.2 Can we use CLIP to resolve spatial relations?

A key limitation in Isolated Proposal Scoring is that relations between objects in different proposals are not taken into account. For example, in Figure 2, the information about the spatial relationships among the cats is lost when the proposals are isolated. In order to use CLIP to decide which object has a specified relation to another object, the model's output must encode the spatial relation in question. Therefore, we design an experiment to determine whether a pre-trained model, such as CLIP, can understand spatial relations within the context of its pre-training task. We generate synthetic images using the process described for the CLEVR dataset (Johnson et al., 2017). These scenes include three shapes–spheres, cubes, and cylinders–and eight colors–gray, blue, green, cyan, yellow, purple, brown, red.

Model	Text-pair Spatial	Text-pair Non-spatial	Image-pair Spatial	Image-pair Non-spatial
CLIP RN50x4	43.73	89.83	48.90	97.36
CLIP RN50x16	52.54	90.17	49.78	96.48
CLIP ViT-B/32	48.81	95.25	48.90	96.48
CLIP ViT-B/16	50.51	92.88	50.22	97.36

Table 1: Accuracy on CLEVR image-text matching task. CLIP performs well on the non-spatial version of the task but poorly on the spatial version. Text-pair tasks have 295 instances each; image-pair tasks have 227 instances each.

In the *text-pair* version of our tasks, using the object attribute and position information associated with each image, we randomly select one of the pairwise relationships between objects–left, right, front, or behind–and construct a sentence fragment based on it. For example: "A blue sphere to the left of a red cylinder." We also write a distractor fragment that replaces the relation with its opposite. In this case, the distractor would be "A blue sphere to the right of a red cylinder." The task, similar to the contrastive and image-text matching tasks used to pre-train these models, is to choose the correct sentence given the image. As a reference point, we also evaluate on a control (non-spatial) task in which the correct text is a list of the scene's objects and the distractor text is identical except that one object is swapped with a random object not in the scene. For example, if the correct text is "A blue sphere and a red cylinder," then the distractor text could be "A blue sphere and a blue cylinder."

In the *image-pair* version of our tasks, we have a single sentence fragment constructed as described above for the spatial and control (non-spatial) tasks and two images such that only one matches the text. Appendix B shows examples of these tasks.

CLIP's performance on these tasks is shown in Table 1. Similar results for the pre-trained model ALBEF (Li et al., 2021) are shown in Appendix D.1 While performance on the control task is quite good, accuracy on the spatial task is not so dif-

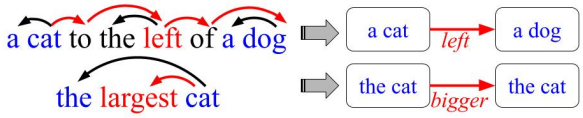


Figure 3: Example extraction of semantic trees from dependency parses. Predicate text in blue. Red arcs show paths contributing spatial relation *left* and superlative *largest*. For the superlative, we create a parent node with the original node as the only child, effectively converting it into a relation.

ferent from random chance (50%). This indicates that the model scores of image-text pairs largely do not take spatial relations into account.

3.3 Spatial Relation Resolver

Since CLIP lacks sensitivity to spatial relations, we propose to decompose complex expressions into simpler primitives. The basic primitive is a predicate applying to an object, which we use CLIP to answer. The second primitive is a spatial relation between objects, for which we use heuristic rules.

Predicates A predicate is a textual property that the referent must satisfy. For example, "the cat" and "blue airplane" are predicates. We write $P(i)$ to say that object i satisfies the predicate P. We model P as a categorical distribution over objects, and estimate $p(i) = \Pr[P(i)]$ with the pre-trained model using isolated proposal scoring (§ 3.1).

Relations We have already discussed the importance of binary spatial relations like "the cat to the *left* of the dog" for the ReC task. We consider seven spatial relations–*left*, *right*, *above*, *below*, *bigger*, *smaller*, and *inside*. We write $R(i, j)$ to mean that the relation R holds between objects i and j, and we use heuristics to determine the probability $r(i, j) = \Pr[R(i, j)]$. For example, for *left*, we set $r(i, j) = 1$ if the center point of box i is to the left of the center point of box j and $r(i, j) = 0$ otherwise. §C.1 describes all relation semantics.

Superlative Relations We also consider superlatives, which refer to an object that has some relation to all other objects satisfying the same predicate, e.g. "leftmost dog". We handle superlatives as a special case of relations where the empty second argument is filled by copying the predicate specifying the first argument. Thus, "leftmost dog" effectively finds the dog that is most likely to the left of other dog(s). Our set of superlative relation types is the same as our set of relation types, excluding *inside*.

Semantic Trees Having outlined the semantic formalism underlying our method, we can describe it procedurally. We first use spaCy (Honnibal and Johnson, 2015) to build a dependency parse for the expression. As illustrated in Figure 3, we extract a semantic tree from the dependency parse, where each noun chunk becomes a node, and dependency paths between the heads of noun chunks become relations between entities based on the keywords they contain. See §C.2 for extraction details.

In the tree, each node N contains a predicate P_N and has a set of children; an edge (N, N') between N and its child N' corresponds to a relation $R_{N,N'}$. For example, as shown in Figure 3, "a cat to the left of a dog" would be parsed as a node containing the predicate "a cat" connected by the relation *left* to its child corresponding to "a dog". We define $\pi_N(i)$ as the probability that node N refers to object i, and compute it recursively. For each node N, we first set $\pi_N(i) = p_N(i)$ and then iterate through each child N' and update $\pi_N(i)$ as follows[6]:

$$\pi'_N(i) \propto \pi_N(i) \sum_j \Pr\left[R_{N,N'}(i, j) \wedge P_{N'}(j)\right]$$

$$\propto \pi_N(i) \sum_j r_{N,N'}(i, j)\pi_{N'}(j).$$

The last line makes the simplifying assumption that all predicates and relations are independent.[7]

To compute our final output, we ensemble the distribution π_{root} for the root node with the output of plain isolated proposal scoring (with the whole input expression) by multiplying the proposal probabilities elementwise. This method gives us a principled way to combine predicates (P_N) with spatial relational constraints ($R_{N,N'}$) for each node N.

4 Experiments

4.1 Datasets

We compare ReCLIP to other zero-shot methods on **RefCOCOg** (Mao et al., 2016), **RefCOCO** and **RefCOCO+** (Yu et al., 2016). These datasets use images from MS COCO (Lin et al., 2014). RefCOCO and RefCOCO+ were created in a two-player game, and RefCOCO+ is designed to avoid spatial relations. RefCOCOg includes spatial relations and has longer expressions on average. For comparing zero-shot methods with the out-of-domain performance of models trained on COCO, we use **RefGTA** (Tanaka et al., 2019), which contains images

[6] Superlatives of a node are processed after all its relations.

[7] We write $\propto$ because $\pi'_N(i)$ is normalized to sum to 1.

Model	RefCOCOg Val	RefCOCOg Test	RefCOCO+ Val	RefCOCO+ TestA	RefCOCO+ TestB	RefCOCO Val	RefCOCO TestA	RefCOCO TestB
Random	18.12	19.10	16.29	13.57	19.60	15.73	13.51	19.20
Supervised SOTA	83.35	81.64	81.13	85.52	72.96	87.51	90.40	82.67
CPT-Blk w/ VinVL (Yao et al., 2021)	32.1	32.3	25.4	25.0	27.0	26.9	27.5	27.4
CPT-Seg w/ VinVL (Yao et al., 2021)	36.7	36.5	31.9	35.2	28.8	32.2	36.1	30.3
CLIP								
CPT-adapted	22.26	23.66	23.76	21.57	25.92	23.12	21.46	26.93
GradCAM	50.96	49.72	**47.81**	**56.90**	37.72	42.91	**51.05**	35.23
ReCLIP w/o relations	57.76	57.13	47.43	50.05	43.91	41.97	43.45	39.98
ReCLIP	**60.48**	**59.74**	46.92	48.83	**45.00**	**46.03**	46.17	**47.38**
CLIP w/ Object Size Prior								
CPT-adapted	28.98	30.12	26.59	25.25	27.20	26.11	25.35	28.09
GradCAM	52.35	51.27	49.40	59.59	38.64	44.66	53.47	36.21
ReCLIP w/o relations	59.25	58.95	54.51	60.18	46.23	48.57	53.63	40.77
ReCLIP	62.03	61.88	54.66	59.68	47.43	54.73	58.94	50.25

Table 2: Accuracy on the RefCOCOg, RefCOCO+ and RefCOCO datasets. ReCLIP outperforms other zero-shot methods on RefCOCOg. On RefCOCO+ and RefCOCO, ReCLIP is on par with or better than GradCAM on average and has lower variance between TestA and TestB, which correspond to different kinds of objects. When taking into account a prior on object size (filtering out objects smaller than 5% of the image), GradCAM's advantage on the TestA splits is erased. Best zero-shot results in each column are in **bold**, and best zero-shot results using the size prior are underlined. CLIP results use an ensemble of the RN50x16 and ViT-B/32 CLIP models. CPT-adapted is an adapted version of CPT-Blk. Supervised SOTA refers to MDETR (Kamath et al., 2021); we use the EfficientNet-B3 version. All methods except MDETR use detected proposals from MAttNet (Yu et al., 2018). CPT-Seg uses Mask-RCNN segmentation masks from Yu et al. (2018).

from the Grand Theft Auto video game. All referring expressions in RefGTA correspond to people, and the objects (i.e. people) tend to be much smaller on average than those in RefCOCO/g/+.

4.2 Implementation Details

We use an ensemble of the CLIP RN50x16 and ViT-B/32 models (results for individual models are shown in Appendix F). GradCAM's hyperparameter α controls the effect of the proposal's area on its score. We select $\alpha = 0.5$ for all models based on tuning on the RefCOCOg validation set. We emphasize that the optimal value of α for a dataset depends on the size distribution of ground-truth objects. ReCLIP also has a hyperparameter, namely the standard deviation σ. We try a few values on the RefCOCOg validation set and choose $\sigma = 100$, as we show in Appendix E.4, isolated proposal scoring has little sensitivity to σ. As discussed by (Perez et al., 2021), zero-shot experiments often use labeled data for model selection. Over the course of this work, we primarily experimented with the RefCOCOg validation set and to a lesser extent with the RefCOCO+ validation set. For isolated proposal scoring, the main variants explored are documented in our ablation study (§4.6). Other techniques that we tried, including for relation-handling, and further implementation details are given in Appendix E.

4.3 Results on RefCOCO/g/+

Table 2 shows results on RefCOCO, RefCOCO+, and RefCOCOg. ReCLIP is better than the other zero-shot methods on RefCOCOg and RefCOCO and on par with GradCAM on RefCOCO+. However, GradCAM has a much higher variance in its accuracy between the TestA and TestB splits of RefCOCO+ and RefCOCO. We note that GradCAM's hyperparameter α, controlling the effect of proposal size, was tuned on the RefCOCOg validation set, and RefCOCOg was designed such that boxes of referents are at least 5% of the image area (Mao et al., 2016). In the bottom portion of Table 2, we show that when this 5% threshold, a prior on object size for this domain, is used to filter proposals for both GradCAM and ReCLIP , ReCLIP performs on par with/better than GradCAM on TestA. ReCLIP's spatial relation resolver helps on RefCOCOg and RefCOCO but not on RefCOCO+, which is designed to avoid spatial relations.

4.4 Results on RefGTA

Next, we evaluate on RefGTA to compare our method's performance to the out-of-domain accuracy of two state-of-the-art fully supervised ReC models: UNITER-Large (Chen et al., 2020) and

Model	Val_{gt}	Val_{det}	$Test_{gt}$	$Test_{det}$
Random	27.03	21.53	27.60	21.75
UNITER-$L_{RefCOCO+}$	45.18	42.77	46.09	43.31
UNITER-$L_{RefCOCOg}$	47.15	46.32	47.64	46.87
$MDETR_{RefCOCO+}$	–	38.49	–	39.02
$MDETR_{RefCOCOg}$	–	38.29	–	39.13
$MDETR_{Pretrained}$	–	54.91	–	56.60
CLIP GradCAM	51.89	51.00	51.55	50.70
ReCLIP w/o relations	**71.72**	**70.24**	**72.51**	**70.82**
ReCLIP	70.44	68.78	71.32	69.59

Table 3: Accuracy on RefGTA dataset. ReCLIP w/o relations outperforms all other methods. *gt* denotes use of ground-truth proposals; *det* denotes use of detected proposals. Subscripts *RefCOCO+*/*RefCOCOg* indicate finetuning dataset; *Pretrained* indicates a model that is not finetuned. MDETR does not take proposals as input, so the *gt* columns are blank. We use the EfficientNet-B3 versions of MDETR. **Bold** indicates best score in a column.

MDETR (Kamath et al., 2021).

Like ReCLIP, UNITER takes proposals as input.[8] We show results using ground-truth proposals and detections from UniDet (Zhou et al., 2021), which is trained on the COCO, Objects365 (Shao et al., 2019), OpenImages (Kuznetsova et al., 2020), and Mapillary (Neuhold et al., 2017) datasets.[9] MDETR does not take proposals as input.

Table 3 shows our results. ReCLIP's accuracy is more than 12% higher than the accuracies of UNITER-Large and MDETR. ReCLIP also outperforms GradCAM by about 20%. The rule-based relation resolver is not helpful on average in this setting. A key reason for this is that all proposals considered are people, and relations in the expressions often involve other objects.

4.5 Using another Pre-trained Model

In order to determine how isolated proposal scoring (IPS) compares to GradCAM and CPT on other pre-trained models, we present results using ALBEF (Li et al., 2021). ALBEF offers two methods for scoring image-text pairs–the output used for its image-text contrastive (ITC) loss and the output used for its image-text matching (ITM) loss. The architecture providing the ITC output is very similar to CLIP–has only a shallow interaction between the image and text modalities. The ITM output is given by an encoder that has deeper interactions between image and text and operates on top of the ITC encoders' output. Appendix D provides more details. The results, shown in Table 4, show that with the ITC output, IPS performs better than GradCAM, but with the ITM output, GradCAM performs better. This suggests that IPS works well across models like CLIP and ALBEF ITC (i.e. contrastively pre-trained with shallow modality interactions) but that GradCAM may be better for models with deeper interactions.

[8]UNITER requires features from the bottom-up top-down attention model (Anderson et al., 2017). We use https://github.com/airsplay/py-bottom-up-attention to compute the features for RefGTA. We note that for RefCOCO+ and RefCOCOg val sets, using features computed from this repository rather than the original features provided by the UNITER authors results in an accuracy decrease of 1.47% (RefCOCO+) and 2.08% (RefCOCOg) when using ground-truth proposals.

[9]For UniDet, we use the confidence threshold of 0.5 suggested by the authors, and filter out the non-person proposals.

Model	RefCOCOg	RefCOCO+(A)	RefCOCO+(B)
ALBEF ITM (Deep modality interaction)			
CPT-adapted	24.99	26.83	26.43
GradCAM	**55.92**	**61.75**	**42.79**
IPS	55.21	51.82	42.63
ALBEF ITC (Shallow modality interaction)			
CPT-adapted	21.10	19.00	21.33
GradCAM	47.53	44.60	36.00
IPS	**54.07**	**45.90**	**39.58**

Table 4: Accuracy on RefCOCOg and RefCOCO+ test sets using ALBEF pre-trained model. IPS does best when using ALBEF's ITC architecture, while GradCAM is better for ITM.

Isolation type	RefCOCOg	RefCOCO+
Crop	54.53	41.24
Blur	56.00	47.29
max(Crop,Blur)	55.84	44.56
Crop+Blur	**57.76**	**47.43**

Table 5: Ablation study of isolation types used to score proposals on Val splits of RefCOCOg/RefCOCO+, using detections from MAttNet (Yu et al., 2018). Crop+Blur is best overall.

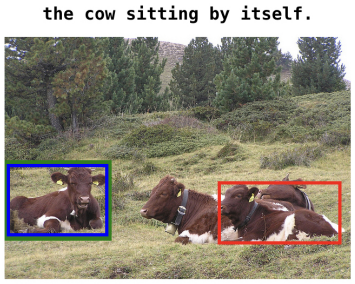
4.6 Analysis

Performance of IPS Our results show that among the region scoring methods that we consider, IPS achieves the highest accuracy for contrastively pre-trained models like CLIP. Figure 4a gives intuition for this—aside from an object's attributes, many referring expressions describe the local context around an object, and IPS focuses on this local context (as well as object attributes).

Table 5 shows that using both cropping and blurring obtains greater accuracy than either alone.

Limitations Although ReCLIP outperforms the baselines that we consider, there is a considerable gap between it and supervised methods. The principal challenge in improving the system is making relation-handling more flexible. There are several object relation types that our spatial relation resolver cannot handle; for instance, those that in-

(a) ReCLIP is correct, while GradCAM is incorrect



(b) Both ReCLIP and GradCAM are incorrect

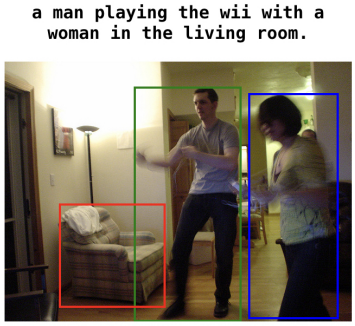


Figure 4: RefCOCOg validation examples. Ground-truth boxes are green, ReCLIP predictions are blue, and GradCAM predictions are red. In 4a, ReCLIP makes the correct prediction based on local context. In 4b, ReCLIP grounds an incorrect noun chunk from the expression.

volve counting: "the second dog from the right." Another challenge is in determining which relations require looking at multiple proposals. For instance, ReCLIP selects a proposal corresponding to the incorrect noun chunk in Figure 4b because the relation resolver has no rule for splitting an expression on the relation "with." Depending on the context, relations like "with" may or may not require looking at multiple proposals, so handling them is challenging for a rule-based system.

5 Related Work

Referring expression comprehension Datasets for ReC span several visual domains, including photos of everyday scenes (Mao et al., 2016; Kazemzadeh et al., 2014), video games (Tanaka et al., 2019), objects in robotic context (Shridhar et al., 2020; Wang et al., 2021), and webpages (Wichers et al., 2018). Spatial heuristics have been used in previous work (Moratz and Tenbrink, 2006). There is a long line of work in weakly supervised ReC, where at training time, pairs of referring expressions and images are available but the ground-truth bounding boxes for each expression are not (Rohrbach et al., 2016; Liu et al., 2019; Zhang et al., 2018, 2020; Sun et al., 2021). Our setting differs from the weakly supervised setting in that the model is not trained at all on the ReC task. Sadhu et al. (2019) discuss a zero-shot setting different from ours in which novel objects seen at test time, but the visual domain stays the same.

Pre-trained vision and language models Early pre-trained vision and language models (Tan and Bansal, 2019; Lu et al., 2019; Chen et al., 2020) used a cross-modal transformer (Vaswani et al., 2017) and pre-training tasks like masked language modeling, image-text matching, and image feature regression. By contrast, CLIP and similar models (Radford et al., 2021; Jia et al., 2021) use a separate image and text transformer and a contrastive pre-training objective. Recent hybrid approaches augment CLIP's architecture with a multi-modal transformer (Li et al., 2021; Zellers et al., 2021).

Zero-shot application of pre-trained models Models pre-trained with the contrastive objective have exhibited strong zero-shot performance in image classification tasks (Radford et al., 2021; Jia et al., 2021). Gu et al. (2021) use CLIP can be to classify objects by computing scores for class labels with cropped proposals. Our IPS is different in that it isolates proposals by both cropping *and blurring*. Shen et al. (2021) show that a simple zero-shot application of CLIP to visual question answering performs almost on par with random chance. Yao et al. (2021) describe a zero-shot method for ReC based on a pre-trained masked language model (MLM); we show that their zero-shot results and a version of their method adapted for models pre-trained to compute image-text scores (rather than MLM) are substantially worse than isolated proposal scoring and GradCAM.

6 Conclusion

We present ReCLIP, a zero-shot method for referring expression comprehension (ReC) that decomposes an expression into subqueries, uses CLIP to score isolated proposals against these subqueries, and combines the outputs with spatial heuristics. ReCLIP outperforms zero-shot ReC approaches from prior work and also performs well across visual domains: ReCLIP outperforms state-of-the-art supervised ReC models, trained on natural images, when evaluated on RefGTA. We also find that CLIP has low zero-shot spatial reasoning performance, suggesting the need for pre-training methods that account more for spatial reasoning.

7 Ethical and Broader Impacts

Recent work has shown that pre-trained vision and language models suffer from biases such as gender bias (Ross et al., 2021; Srinivasan and Bisk, 2021). Given that CLIP was trained on data collected from the web and not necessarily curated carefully, CLIP could suffer from such biases as well. Therefore, we do not advise deploying our system directly in the real world immediately. Instead, practitioners interested in this system should first perform analysis to measure its biases based on previous work and attempt to mitigate them. We also note that our work relies heavily on a pre-trained model whose pre-training required a great deal of energy, which likely had negative environmental effects. That being said our zero-shot method does not require training a new model and in that sense could be more environmentally friendly than supervised ReC models (depending on the difference in the cost of inference).

References

Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2017. Bottom-up and top-down attention for image captioning and vqa. *ArXiv*, abs/1707.07998.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. 2020. Uniter: Universal image-text representation learning. In *ECCV*.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*.

Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. 2021. Zero-shot detection via vision and language knowledge distillation. *CoRR*, abs/2104.13921.

Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778.

Matthew Honnibal and Mark Johnson. 2015. An improved non-monotonic transition system for dependency parsing. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1373–1378, Lisbon, Portugal. Association for Computational Linguistics.

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

Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C. Lawrence Zitnick, and Ross B. Girshick. 2017. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1988–1997.

Aishwarya Kamath, Mannat Singh, Yann LeCun, Gabriel Synnaeve, Ishan Misra, and Nicolas Carion. 2021. Mdetr - modulated detection for end-to-end multi-modal understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1780–1790.

Jared Kaplan, Sam McCandlish, T. J. Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeff Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *ArXiv*, abs/2001.08361.

Sahar Kazemzadeh, Vicente Ordonez, Marc André Matten, and Tamara L. Berg. 2014. Referitgame: Referring to objects in photographs of natural scenes. In *EMNLP*.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Li Fei-Fei. 2016. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123:32–73.

Alina Kuznetsova, Hassan Rom, Neil Gordon Alldrin, Jasper R. R. Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, Tom Duerig, and Vittorio Ferrari. 2020. The open images dataset v4. *International Journal of Computer Vision*, 128:1956–1981.

Junnan Li, Ramprasaath R. Selvaraju, Akhilesh Deepak Gotmare, Shafiq Joty, Caiming Xiong, and Steven Hoi. 2021. Align before fuse: Vision and language representation learning with momentum distillation. In *NeurIPS*.

Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *ECCV*.

Xuejing Liu, Liang Li, Shuhui Wang, Zhengjun Zha, Dechao Meng, and Qingming Huang. 2019. Adaptive reconstruction network for weakly supervised referring expression grounding. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2611–2620.

Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *NeurIPS*.

Junhua Mao, Jonathan Huang, Alexander Toshev, Oana-Maria Camburu, Alan Loddon Yuille, and Kevin P. Murphy. 2016. Generation and comprehension of unambiguous object descriptions. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11–20.

Reinhard Moratz and Thora Tenbrink. 2006. Spatial reference in linguistic human-robot interaction: Iterative, empirically supported development of a model of projective relations. *Spatial cognition and computation*, 6(1):63–107.

Gerhard Neuhold, Tobias Ollmann, Samuel Rota Bulò, and Peter Kontschieder. 2017. The mapillary vistas dataset for semantic understanding of street scenes. In *International Conference on Computer Vision (ICCV)*.

Vicente Ordonez, Girish Kulkarni, and Tamara L. Berg. 2011. Im2text: Describing images using 1 million captioned photographs. In *Neural Information Processing Systems (NIPS)*.

Ethan Perez, Douwe Kiela, and Kyunghyun Cho. 2021. True few-shot learning with language models. *ArXiv*, abs/2105.11447.

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

Anna Rohrbach, Marcus Rohrbach, Ronghang Hu, Trevor Darrell, and Bernt Schiele. 2016. Grounding of textual phrases in images by reconstruction. *ECCV*.

Candace Ross, Boris Katz, and Andrei Barbu. 2021. Measuring social biases in grounded vision and language embeddings. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 998–1008, Online. Association for Computational Linguistics.

Arka Sadhu, Kan Chen, and Ramakant Nevatia. 2019. Zero-shot grounding of objects from natural language queries. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4693–4702.

Ramprasaath R. Selvaraju, Abhishek Das, Ramakrishna Vedantam, Michael Cogswell, Devi Parikh, and Dhruv Batra. 2019. Grad-cam: Visual explanations from deep networks via gradient-based localization. *International Journal of Computer Vision*, 128:336–359.

Shuai Shao, Zeming Li, Tianyuan Zhang, Chao Peng, Gang Yu, Xiangyu Zhang, Jing Li, and Jian Sun. 2019. Objects365: A large-scale, high-quality dataset for object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

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

Sheng Shen, Liunian Harold Li, Hao Tan, Mohit Bansal, Anna Rohrbach, Kai-Wei Chang, Zhewei Yao, and Kurt Keutzer. 2021. How much can clip benefit vision-and-language tasks? *ArXiv*, abs/2107.06383.

Mohit Shridhar, Dixant Mittal, and David Hsu. 2020. Ingress: Interactive visual grounding of referring expressions. *The International Journal of Robotics Research*, 39:217 – 232.

Tejas Srinivasan and Yonatan Bisk. 2021. Worst of both worlds: Biases compound in pre-trained vision-and-language models. *arXiv preprint arXiv:2104.08666*.

Mingjie Sun, Jimin Xiao, Eng Gee Lim, Si Liu, and John Yannis Goulermas. 2021. Discriminative triad matching and reconstruction for weakly referring expression grounding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43:4189–4195.

Hao Hao Tan and Mohit Bansal. 2019. Lxmert: Learning cross-modality encoder representations from transformers. In *EMNLP*.

Mikihiro Tanaka, Takayuki Itamochi, Kenichi Narioka, Ikuro Sato, Y. Ushiku, and Tatsuya Harada. 2019. Generating easy-to-understand referring expressions for target identifications. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5793–5802.

Ashish Vaswani, Noam M. Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *ArXiv*, abs/1706.03762.

Ke-Jyun Wang, Yun-Hsuan Liu, Hung-Ting Su, Jen-Wei Wang, Yu-Siang Wang, Winston H. Hsu, and Wen-Chin Chen. 2021. Ocid-ref: A 3d robotic dataset with embodied language for clutter scene grounding. In *NAACL*.

Nevan Wichers, Dilek Z. Hakkani-Tür, and Jindong Chen. 2018. Resolving referring expressions in images with labeled elements. *2018 IEEE Spoken Language Technology Workshop (SLT)*, pages 800–806.

Yuan Yao, Ao Zhang, Zhengyan Zhang, Zhiyuan Liu, Tat-Seng Chua, and Maosong Sun. 2021. Cpt: Colorful prompt tuning for pre-trained vision-language models. *ArXiv*, abs/2109.11797.

Licheng Yu, Zhe L. Lin, Xiaohui Shen, Jimei Yang, Xin Lu, Mohit Bansal, and Tamara L. Berg. 2018. Mattnet: Modular attention network for referring expression comprehension. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1307–1315.

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

Rowan Zellers, Ximing Lu, Jack Hessel, Youngjae Yu, Jae Sung Park, Jize Cao, Ali Farhadi, and Yejin Choi. 2021. Merlot: Multimodal neural script knowledge models. *NeurIPS*.

Hanwang Zhang, Yulei Niu, and Shih-Fu Chang. 2018. Grounding referring expressions in images by variational context. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4158–4166.

Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. 2021. Vinvl: Revisiting visual representations in vision-language models. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5575–5584.

Zhu Zhang, Zhou Zhao, Zhijie Lin, Jieming Zhu, and Xiuqiang He. 2020. Counterfactual contrastive learning for weakly-supervised vision-language grounding. In *NeurIPS*.

Xingyi Zhou, Vladlen Koltun, and Philipp Krähenbühl. 2021. Simple multi-dataset detection. *ArXiv*, abs/2102.13086.

Figure 5: The visual representation of a proposal using CPT-adapted. The example is taken from the RefCOCOg validation set.

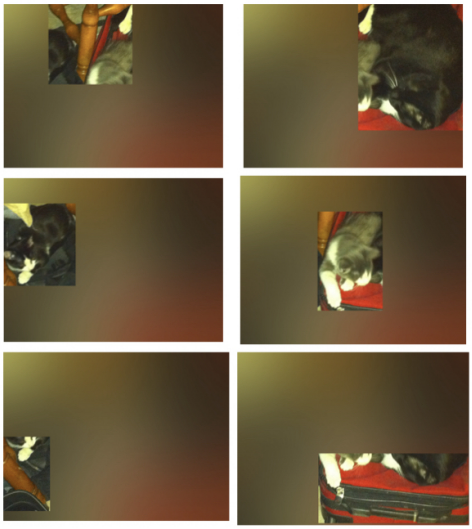


Figure 6: An example of isolating proposals by blurring the remainder of the image using $\sigma = 100$

A Visualization of Region-Scoring Methods

A.1 Colorful Prompt Tuning (CPT)

Figure 5 shows an example of the visual representation of a proposal using CPT-adapted.

A.2 Isolated Proposal Scoring (IPS)

Figure 6 shows the blurred versions of the proposals for an image using $\sigma = 100$.

B Synthetic Spatial Reasoning Experiment

Figure 7 gives an example of the *text-pairs* version of the synthetic tasks.

Figure 8 gives an example of the *image-pairs* version of the synthetic tasks.

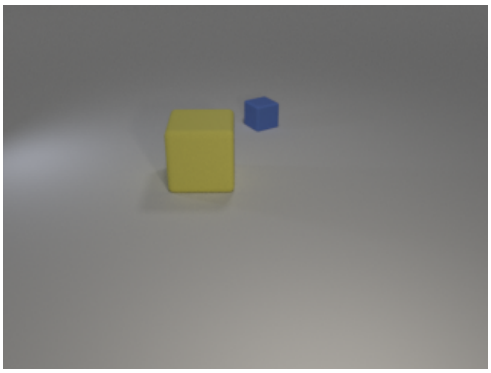
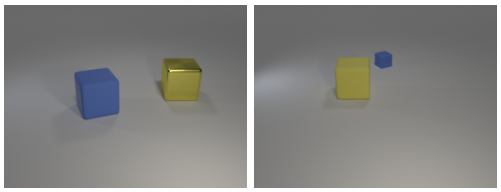
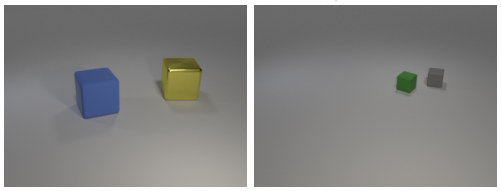


Figure 7: Example image for the synthetic text-pair tasks. For the spatial task, the text pair corresponding to this image is "a yellow cube is in front of a blue cube." (correct) and "a yellow cube is behind a blue cube." (incorrect). For the non-spatial (control) task, the text pair corresponding to this image is "a blue cube and a yellow cube" (correct) and "a blue cube and a yellow sphere" (incorrect).



(a) "a blue cube to the left of a yellow cube."



(b) "a blue cube and a yellow cube"

Figure 8: Examples of the image-pairs version of the spatial (8a) and non-spatial (8b) tasks. In each case, the left image is the correct one.

C Semantic Formalism

C.1 Relation Semantics

We use deterministic heuristics to compute the semantics of the following six relations: *left*, *right*, *above*, *below*, *bigger*, and *smaller*. On the other hand, we treat *inside* as a random variable, and use heuristics to compute the value of its parameter.

For $R \in \{left, right, above, below\}$, we compute $R(i, j)$ by checking whether R holds between the center point of box i and box j. For example, if the center point of i is to the left of the center point of box j, then $left(i, j) = 1$.

We compute $bigger(i, j)$ and $smaller(i, j)$ simply by comparing the areas of boxes i and j. For example, $bigger(i, j)$ checks that the area of box i is greater than the area of box j.

Finally, for $R = inside$, we parameterize $r(i, j)$ as the ratio between the are of the intersection of boxes i, j compared to the area of box i. Thus, unlike the other six deterministic rules, *inside* is modeled as a random variable.

C.2 Relation Extraction

We identify noun chunks in the dependency parse as predicates. We then extract relations by looking for dependency paths between the heads of noun chunks that contain the following keywords:

- *left*: "left", "west"
- *right*: "right", "east"
- *above*: "above", "north", "top", "back", "behind"
- *below*: "below", "south", "under", "front"
- *bigger*: "bigger", "larger", "closer"
- *smaller*: "smaller", "tinier", "further"
- *inside*: "inside", "within", "contained"

We extract superlative relations by looking for dependency paths off the head of a noun chunk containing the following keywords:

- *left*: "left", "west", "leftmost", "western"
- *right*: "right", "rightmost", "east", "eastern"
- *above*: "above", "north", "top"
- *below*: "below", "south", "underneath", "front"
- *bigger*: "bigger", "biggest", "larger", "largest", "closer", "closest"
- *smaller*: "smaller", "smallest", "tinier", "tiniest", "further", "furthest"

D Description of ALBEF

The ALBEF model has an image-only transformer and a text-only transformer like CLIP but also has a multi-modal transformer that operates on the outputs of these two transformers. ALBEF is pretrained with three losses: (1) an image-text contrastive (ITC) loss that works just like CLIP's and uses the outputs of the image-only and text-only transformers, (2) an image-text matching (ITM)

Model	Text-pair Spatial	Text-pair Non-spatial	Image-pair Spatial	Image-pair Non-spatial
ALBEF ITM	49.83	92.20	53.74	90.75
ALBEF ITC	49.83	85.42	51.54	72.25

Table 6: Accuracy on CLEVR image-text matching task. ALBEF performs well on the non-spatial version of the task but poorly on the spatial version. Text-pair tasks have 295 instances each; image-pair tasks have 227 instances each.

loss–where the task is to decide whether a given image-text pair match–which uses the outputs of the multi-modal encoder, and (3) a masked language modeling loss which uses the outputs of the multi-modal encoder. We explore both the ITC and ITM scores in our experiments. ALBEF was pre-trained on roughly 15M image-caption pairs from conceptual captions (Sharma et al., 2018), SBU Captions (Ordonez et al., 2011), COCO (Lin et al., 2014), and Visual Genome (Krishna et al., 2016).[10]

D.1 ALBEF Performance on Synthetic Spatial Reasoning Experiment

Table 6 shows the zero-shot accuracy of ALBEF ITM and ITC in the synthetic spatial reasoning experiment described in §3.2.

E Implementation Details

E.1 Text prompt

For ALBEF, we pass the input expression directly to the model, whereas for CLIP, when using GradCAM and ReCLIP (with or without relations), we use the prefix "a photo of" following the authors' observations (Radford et al., 2021). For CPT, the prompt is given in § 2.3.

E.2 Position embeddings

Both CLIP and ALBEF use fixed-size position embeddings, so either the input image must be resized to fit the dimensions of the embeddings or the size of the embeddings must be changed. For all models, we resize the image to match the model's visual input resolution. Resizing of images is done via bicubic interpolation. Figure 9 shows the how the performance of the GradCAM method varies between resizing images and resizing embeddings–for CLIP RN50x16, there is very little difference, while for CLIP ViT-B/32 image resizing makes a larger difference.

[10] As noted by the ALBEF authors, validation/test images of RefCOCO+ and RefCOCOg are included in the training set of COCO.

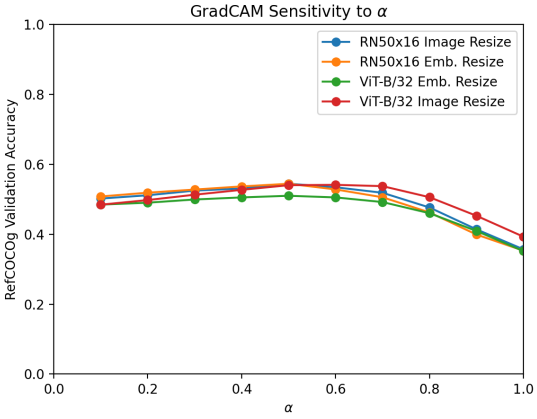


Figure 9: CLIP RN50x16 and ViT-B/32 Performance using GradCAM on RefCOCOg validation set comparing resizing of images with resizing of position embeddings, across 10 values of α. These results use ground-truth proposals.

Hyperparameters Specifically, we evaluate each value in the set $\{0.2, 0.4, 0.6, 0.8, 1.0\}$ and choose the best. The chosen values are $\alpha = 0.8$ for CLIP RN50x16 and ALBEF ITC and $\alpha = 1.0$ for CLIP ViT-B/32.

E.3 GradCAM Layer

For CLIP ViT-B/32, we use the last layer of the visual transformer for GradCAM. For CLIP RN50x16, we use output of layer 4 for GradCAM. For ALBEF ITM, we use the third layer of the multi-modal transformer for GradCAM (following Li et al. (2021)). For ALBEF ITC, we use the final layer of the visual transformer for GradCAM.

E.4 Hyperparameter sensitivity

Figure 9 shows the sensitivity of the GradCAM method to α for the two CLIP models. We choose $\alpha = 0.5$ for all models (including ALBEF), which results in the best accuracy for almost models. For ViT-B/32, $\alpha = 0.6$ yields slightly higher accuracy by (0.1%) on the RefCOCOg validation set. Figure 10 shows the sensitivity of the IPS method to the blur standard deviation σ for the CLIP RN50x16 model. As shown, the method has little sensitivity to σ above $\sigma = 20$.

E.5 Experimentation on validation set

As discussed by Perez et al. (2021), research on the zero-shot setting often uses labeled data for model selection. Aside from variants of IPS documented in our ablation study (§4.6), we also experimented

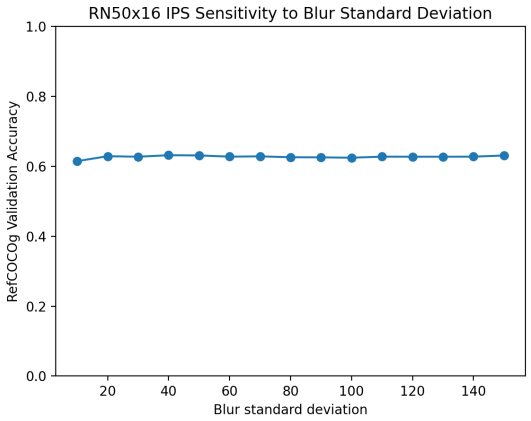


Figure 10: CLIP RN50x16 Performance using IPS on RefCOCOg validation set for different values of blur standard deviation σ. These results use ground-truth proposals.

on the RefCOCOg validation set (and to a lesser extent on the RefCOCO+ validation set) with:

1. Drawing a rectangle around the proposal and using an appropriate text prompt. Performance was somewhat similar to CPT performance.

2. Ensembling the original text prompt with a text prompt having only the noun chunk of the expression containing the head word. This helped for IPS and is in a sense part of our rule-based relation-handling.

3. Other techniques for handling superlatives. For instance, we tried to compute $\Pr[P_N(i) \wedge \bigwedge_{j \neq i}(\neg P_N(j) \vee (P_N(j) \wedge R(i,j)))]$. This performed worse than our chosen technique on the RefCOCOg validation set.

Most of these preliminary experiments were performed using the area threshold mentioned in §4.3.

E.6 Description of Computing Infrastructure

We primarily used a machine with Quadro RTX 8000 GPUs and Google Cloud machines with V100 GPUs. These machines used Ubuntu as the operating system.

E.7 Dataset Information

All datasets that we use are focused on English. The COCO dataset can be downloaded from https://cocodataset.org/#download. The RefCOCO/g/+ datasets can be downloaded from https://github.com/lichengunc/refer/tree/master/data. The RefGTA dataset can be downloaded from https://github.com/mikittt/easy-to-understand-REG/tree/master/pyutils/refer2. The RefCOCOg validation set has 4896 instances, the RefCOCOg test set has 9602 instances, the RefCOCO+ validation set has 10758 instances, the RefCOCO+ TestA set has 5726 instances, the RefCOCO+ TestB set has 4889 instances, the RefCOCO validation set has 10834 instances, the RefCOCO TestA set has 5657 instances, the RefCOCO TestB set has 5095 instances, the RefGTA validation set has 17766 instances, and the RefGTA test set has 17646 instances.

F Additional Experiment Results

Table 7 shows full results on the RefCOCOg and RefCOCO+ datasets. Table 8 shows full results on the RefCOCO dataset.

Model	RefCOCOg				RefCOCO+					
	Val$_g$	**Val**$_d$	**Test**$_g$	**Test**$_d$	**Val**$_g$	**Val**$_d$	**TestA**$_g$	**TestA**$_d$	**TestB**$_g$	**TestB**$_d$
Random	20.18	18.117	20.34	19.10	16.73	16.29	12.57	13.57	22.13	19.60
UNITER-L (supervised; Chen et al. (2020))	87.85	74.86	87.73	75.77	84.25	75.90	86.34	81.45	79.75	75.77
MDETR (supervised; Kamath et al. (2021))	–	83.35	–	81.64	–	81.13	–	85.52	–	72.96
Weakly supervised (non-pretrained; Sun et al. (2021))	–	–	–	–	39.18	38.91	40.01	39.91	38.08	37.09
CPT-Blk w/ VinVL (Yao et al., 2021)	–	32.1	–	32.3	–	25.4	–	25.0	–	27.0
CPT-Seg w/ VinVL (Yao et al., 2021)	–	36.7	–	36.5	–	31.9	–	35.2	–	28.8
CLIP RN50x16										
CPT-adapted	27.74	25.10	28.82	26.04	24.43	22.14	20.26	19.54	27.78	25.63
GradCAM	54.47	48.35	53.71	47.47	**48.27**	**44.59**	**52.81**	**52.71**	41.17	35.63
ReCLIP w/o relations	62.46	55.94	62.00	54.35	47.06	44.06	46.49	45.98	49.44	41.87
ReCLIP	**65.32**	**57.84**	**65.10**	**56.56**	46.87	43.44	45.02	44.69	**50.50**	**42.77**
CLIP ViT-B/32										
CPT-adapted	24.10	21.90	24.76	22.78	25.08	23.44	22.27	21.71	28.57	26.24
GradCAM	54.00	49.55	54.01	48.57	48.01	44.65	**52.15**	**50.77**	43.77	39.03
ReCLIP w/o relations	62.40	55.33	61.78	54.35	48.61	**45.05**	50.21	48.25	47.21	41.56
ReCLIP	**66.14**	**57.88**	**64.84**	**56.88**	**48.71**	44.93	49.56	47.66	**48.66**	**42.61**
CLIP Ensemble										
CPT-adapted	26.02	22.26	25.79	23.66	25.52	23.76	21.95	21.57	29.98	25.92
GradCAM	56.94	50.96	56.23	49.72	51.06	**47.81**	**57.82**	**56.90**	43.16	37.72
ReCLIP w/o relations	65.26	57.76	64.64	57.13	51.56	47.43	51.76	50.05	50.93	43.91
ReCLIP	**68.61**	**60.48**	**67.89**	**59.74**	**51.63**	46.92	50.26	48.83	**51.81**	**45.00**

Table 7: Accuracy on the RefCOCOg and RefCOCO+ datasets. ReCLIP outperforms other zero-shot methods on RefCOCOg. On RefCOCO+, ReCLIP is roughly on par with GradCAM but has lower variance between TestA and TestB, which correspond to different kinds of objects. Subscript *g* indicates ground-truth proposals are used, and *d* indicates detected proposals are used. Best zero-shot results for each model and each column are in **bold**. See Table 2 for results using object size prior.

Model	RefCOCO					
	Val$_g$	**Val**$_d$	**TestA**$_g$	**TestA**$_d$	**TestB**$_g$	**TestB**$_d$
Random	16.37	15.73	12.45	13.51	21.32	19.20
UNITER-L (supervised; Chen et al. (2020))	91.84	81.41	92.65	87.04	91.19	74.17
MDETR (supervised; Kamath et al. (2021))	–	87.51	–	90.40	–	82.67
Weakly supervised (non-pretrained; Sun et al. (2021))	39.21	38.35	41.14	39.51	37.72	37.01
CPT-Blk w/ VinVL (Yao et al., 2021)	–	26.9	–	27.5	–	27.4
CPT-Seg w/ VinVL (Yao et al., 2021)	–	32.2	–	36.1	–	30.3
CLIP RN50x16						
CPT-adapted	23.35	21.47	19.30	18.68	28.38	25.28
GradCAM	44.01	40.47	**47.32**	**46.46**	38.12	33.70
ReCLIP w/o relations	40.59	37.63	39.14	38.45	43.53	37.04
ReCLIP	**46.22**	**41.51**	41.45	40.73	**53.27**	**46.05**
CLIP ViT-B/32						
CPT-adapted	25.22	23.71	23.28	22.77	28.40	25.99
GradCAM	45.38	42.30	**50.15**	**49.09**	41.55	36.62
ReCLIP w/o relations	44.35	40.60	45.04	44.02	43.49	37.57
ReCLIP	**49.93**	**45.85**	48.24	47.34	**52.72**	**46.28**
CLIP Ensemble						
CPT-adapted	24.82	23.12	21.69	21.46	28.99	26.93
GradCAM	46.67	42.91	**51.86**	**51.05**	40.10	35.23
ReCLIP w/o relations	45.65	41.97	45.11	43.45	45.50	39.98
ReCLIP	**50.83**	**46.03**	47.29	46.17	**55.96**	**47.38**

Table 8: Accuracy on the RefCOCO dataset. Subscript *g* indicates ground-truth proposals are used, and *d* indicates detected proposals are used. Best zero-shot results for each model and each column are in **bold**. See Table 2 for results using object size prior.