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

Sanjay Subramanian\textsuperscript{1}  William Merrill\textsuperscript{2}  Trevor Darrell\textsuperscript{1}  Matt Gardner\textsuperscript{3}  Sameer Singh\textsuperscript{4,5}  Anna Rohrbach\textsuperscript{1}

\textsuperscript{1}UC Berkeley  \textsuperscript{2}New York University  \textsuperscript{3}Microsoft Semantic Machines  \textsuperscript{4}UC Irvine  \textsuperscript{5}Allen Institute for AI (AI2)

{sanjayss,trevordarrell,anna.rohrbach}@berkeley.edu, wcm9940@nyu.edu, mattgardner@microsoft.com, sameer@uci.edu

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 29% on RefCOCOg, and on RefGTA (video game imagery), ReCLIP’s relative improvement over supervised ReC models trained on real images is 8%.

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.

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 addi-
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 $b_4$.

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 about 20%. Compared to a stronger GradCAM (Selvaraju et al., 2017) 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 4.5% (relative improvement of 8%).

In summary, our contributions include: (1) ReCLIP, a simple but strong new baseline for zero-shot ReC.
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.

2 Background

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

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 $x \in \mathbb{R}^d$. 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 \[
\exp(\beta x_i^T y_j) / \sum_{k=1}^{N} \exp(\beta x_i^T y_k), \]
where $\beta$ 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., 2017). 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, and Chefer et al. (2021). 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

\(^1\)Our code is available at https://www.github.com/allenai/reclip.

\(^2\)This dataset is not public.

\(^3\)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.

\(^4\)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.
The convolutional version, following Selvaraju et al. (2017), applies global average pooling to the gradients, unlike

### 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.

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

<table>
<thead>
<tr>
<th>Model</th>
<th>Text-pair Spatial</th>
<th>Text-pair Non-spatial</th>
<th>Image-pair Spatial</th>
<th>Image-pair Non-spatial</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLIP RN50x4</td>
<td>43.39</td>
<td>89.83</td>
<td>48.90</td>
<td>97.36</td>
</tr>
<tr>
<td>CLIP RN50x16</td>
<td>51.19</td>
<td>89.83</td>
<td>50.22</td>
<td>96.48</td>
</tr>
<tr>
<td>CLIP RN50x64</td>
<td>47.80</td>
<td>94.58</td>
<td>51.54</td>
<td>97.36</td>
</tr>
<tr>
<td>CLIP ViT-B/32</td>
<td>48.47</td>
<td>95.25</td>
<td>48.90</td>
<td>96.48</td>
</tr>
<tr>
<td>CLIP ViT-B/16</td>
<td>50.51</td>
<td>92.54</td>
<td>50.22</td>
<td>96.92</td>
</tr>
<tr>
<td>CLIP ViT-L/14</td>
<td>52.88</td>
<td>96.27</td>
<td>50.66</td>
<td>94.27</td>
</tr>
</tbody>
</table>

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.
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 different 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.

![Example extraction of semantic trees from dependency parses.](image)

**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.

**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 cases where none of our relation/superlative keywords occur in the text, we simply revert to the plain isolated proposal scoring method using the full text.

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 \( R_{NN'} \). 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\(^7\):

\[
\pi_N(i) \propto \pi_N(i) \sum_j \Pr[R_{NN'}(i, j) \land P_{N'}(j)] \\
\propto \pi_N(i) \sum_j r_{NN'}(i, j) \pi_{N'}(j).
\]

The last line makes the simplifying assumption that all predicates and relations are independent.\(^8\)

To compute our final output, we ensemble the distribution \( \pi_{\text{root}} \) 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 \( (R_{NN'}) \) 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 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 G). We ensemble model outputs by adding together the logits from the two models elementwise before taking the softmax. GradCAM’s hyperparameter $\alpha$ controls the effect of the proposal’s area on its score. We select $\alpha = 0$ for all models based on tuning on the RefCOCOg validation set. We emphasize that the optimal value of $\alpha$ for a dataset depends on the size distribution of ground-truth objects. ReCLIP also has a hyperparameter, namely the standard deviation $\sigma$. 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 $\sigma$. 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 $\alpha$, 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 MDETR (Kamath et al., 2021).

Like ReCLIP, UNITER takes proposals as input.9 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. Following the suggestion of the UniDet authors, we use the confidence threshold of 0.5. MDETR does not take proposals as input.

Table 3 shows our results. For methods that take proposals (all methods except MDETR), we consider two evaluation settings using UniDet–DT-P, in which the detected proposals are filtered to have only proposals whose predicted class label is “person”, and DT, in which all detected proposals are considered. ReCLIP’s accuracy is more than 15% higher than the accuracy of UNITER-Large and roughly 5% more than that of MDETR. ReCLIP also outperforms GradCAM by about 20%, and the gap is larger when all UniDet proposals are considered. ReCLIP w/o relations is 1-2% better than ReCLIP in the settings with ground-truth proposals and filtered UniDet proposals. One possible reason

9UNITER 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 trained UNITER models on RefCOCO+ and RefCOCOg using features computed from this repository. On the RefCOCO+ validation set, the resulting model has an accuracy roughly 0.4% less than that of a model trained and evaluated using the original features (when using ground-truth proposals).
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).

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

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.

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
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.

(a) ReCLIP is correct, while GradCAM is incorrect

(b) Both ReCLIP and GradCAM are incorrect

Figure 4: RefCOCOg validation examples using ground-truth proposals. Ground-truth referents 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.

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

Error Analysis and 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 involve 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.

In the RefCOCO+ validation set, when using detected proposals, there are 75 instances for which ReCLIP answers incorrectly but ReCLIP w/o relations answers correctly. We categorize these instances based on their likely sources of error: 4 instances are ambiguous (multiple valid proposals), in 7 instances the parser misses the head noun chunk, in 14 instances our processing of the parse leads to omissions of text when doing isolated proposal scoring (e.g. in “girl sitting in back,” the only noun chunk is “girl,” so this is the only text used during isolated proposal scoring), 52 cases in which there is an error in the execution of the heuristic (e.g. our spatial definition of a relation does not match the relation in the instance). (There are 2 instances for which we mark 2 categories.) The final category (“execution”) includes several kinds of errors, some examples of which are shown in Appendix F.

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). Our work is also related to Krishnamurthy and Kollar (2013), which similarly decomposes the reasoning process...
into a parsing step and visual execution steps, but the visual execution is driven by learned binary classifiers for each predicate type. In the supervised setting, prior work shows that using an external parser, as we do, leads to lower accuracy than training a language module jointly with the remainder of the model (Hu et al., 2017).

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 are 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 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. Concurrent with our work, Liu et al. (2021) also observe that CLIP has poor zero-shot accuracy when dealing with spatial relations.

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). Agarwal et al. (2021) provide evidence that CLIP has racial and other biases, which makes sense since CLIP was trained on data collected from the web and not necessarily curated carefully. 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).

8 Acknowledgements

We thank the Berkeley NLP group, Medhini Narasimhan, and the anonymous reviewers for helpful comments. We thank Michael Schmitz for help with AI2 infrastructure. This work was supported in part by DoD, including DARPA’s LwLL (FA8750-19-1-0504), and/or SemaFor (HR00112020054) programs, and Berkeley Artificial Intelligence Research (BAIR) industrial alliance programs. Sameer Singh was supported in part by the National Science Foundation grant
#IIS-1817183 and in part by the DARPA MCS program under Contract No. N660011924033 with the United States Office Of Naval Research.

References


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


Melbourne, Australia. Association for Computational Linguistics.


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.

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 $R \in \{\text{left}, \text{right}, \text{above}, \text{below}\}$, 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 $\text{left}(i, j) = 1$.

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

Finally, for \( R = \text{inside} \), we parameterize \( r(i, j) \) as the ratio between the area of the intersection of boxes \( i, j \) compared to the area of box \( i \). Thus, unlike the other six deterministic rules, \textit{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:

- \textit{left}: “left”, “west”
- \textit{right}: “right”, “east”
- \textit{above}: “above”, “north”, “top”, “back”, “behind”
- \textit{below}: “below”, “south”, “under”, “front”
- \textit{bigger}: “bigger”, “larger”, “closer”
- \textit{smaller}: “smaller”, “tinier”, “further”
- \textit{inside}: “inside”, “within”, “contained”

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

- \textit{left}: “left”, “west”, “leftmost”, “western”
- \textit{right}: “right”, “rightmost”, “east”, “eastern”
- \textit{above}: “above”, “north”, “east”
- \textit{below}: “below”, “south”, “underneath”, “front”
- \textit{bigger}: “bigger”, “largest”, “closer”
- \textit{smaller}: “smaller”, “smallest”, “tiniest”, “furthest”
- \textit{inside}: “inside”, “within”, “contained”

### 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 pre-trained 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) 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 ITC 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 Grad-CAM 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.
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 $\alpha$ 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 $\sigma$ for the CLIP RN50x16 model. As shown, the method has little sensitivity to $\sigma$ 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 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 $\Pr[P_N(i) \land \bigwedge_{j\neq i} (\neg P_N(j) \lor (P_N(j) \land R(i,j)))]$. This performed worse than our chosen technique on the RefCOCOg validation set.

4. Invoking the parser and relation-handling pipeline on all sentences rather than only those containing one of the relation/superlative keywords.

We also selected the relation types and keywords based on these validation sets. 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, Google Cloud machines with V100 GPUs, and a machine with TITAN RTX and GeForce 2080s. 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/
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 Qualitative Examples**

Figure 12 shows qualitative examples for the RefGTA validation set. Figure 11 shows examples of the execution errors mentioned in the error analysis in Section 4.6.

**G Additional Experiment Results**

This section presents the full results on the RefCOCOg/RefCOCO+/RefCOCO datasets, including results without ensembling using CLIP RN50x16 and ViT-B/32 models and results using ground-truth proposals. Table 7 shows full results on the RefCOCOg and RefCOCO+ datasets. Table 8 shows full results on the RefCOCO dataset.

![Figure 11: Examples of execution errors causing Re-CLIP to answer incorrectly on instances that it answers correctly when not using the relation-handling method. Parts 11a and 11b show cases where the meaning of “behind” does not match our heuristic, which checks which proposal’s y-coordinate is smaller. Part 11c shows an example where “under” means “directly under.” Part 11d shows an example in which due to the superlative “smallest,” the size of proposals appears to be weighted more heavily by our approach than scores CLIP assigns to the proposals based on the text.](image)
### Table 7: Accuracy on the RefCOCOg and RefCOCO+ datasets.

On RefCOCO+, ReCLIP outperforms other zero-shot methods on RefCOCOg. Best zero-shot results for each model and each column are in bold. See Table 2 for results using object size prior.

<table>
<thead>
<tr>
<th>Model</th>
<th>Val(_g)</th>
<th>Val(_d)</th>
<th>Test(_A)</th>
<th>Test(_A)</th>
<th>Test(_B)</th>
<th>Test(_B)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random</td>
<td>20.18</td>
<td>18.17</td>
<td>16.73</td>
<td>16.29</td>
<td>12.57</td>
<td>13.57</td>
</tr>
<tr>
<td>UNITER-L (supervised; Chen et al. (2020))</td>
<td>87.85</td>
<td>74.86</td>
<td>84.25</td>
<td>75.90</td>
<td>81.45</td>
<td>79.75</td>
</tr>
<tr>
<td>MDETR (supervised; Kamath et al. (2021))</td>
<td>–</td>
<td>83.35</td>
<td>–</td>
<td>81.64</td>
<td>–</td>
<td>85.52</td>
</tr>
<tr>
<td>Weakly supervised (non-pretrained; Sun et al. (2021))</td>
<td>–</td>
<td>–</td>
<td>39.18</td>
<td>38.91</td>
<td>40.01</td>
<td>39.91</td>
</tr>
<tr>
<td>CPT-Blk w/ VinVL (Yao et al., 2021)</td>
<td>–</td>
<td>32.1</td>
<td>–</td>
<td>25.4</td>
<td>–</td>
<td>25.0</td>
</tr>
<tr>
<td>CPT-Seg w/ VinVL (Yao et al., 2021)</td>
<td>–</td>
<td>36.7</td>
<td>–</td>
<td>31.9</td>
<td>–</td>
<td>35.2</td>
</tr>
<tr>
<td>CLIP RN50x16</td>
<td>27.74</td>
<td>25.04</td>
<td>24.48</td>
<td>22.09</td>
<td>20.22</td>
<td>19.54</td>
</tr>
<tr>
<td>ReCLIP w/o relations</td>
<td>62.50</td>
<td>55.88</td>
<td>47.12</td>
<td>44.15</td>
<td>46.47</td>
<td>45.97</td>
</tr>
<tr>
<td>ReCLIP</td>
<td>64.79</td>
<td>57.66</td>
<td>47.92</td>
<td>44.53</td>
<td>46.38</td>
<td>45.88</td>
</tr>
<tr>
<td>CLIP VIT-B/32</td>
<td>24.16</td>
<td>21.77</td>
<td>25.07</td>
<td>23.46</td>
<td>22.28</td>
<td>21.73</td>
</tr>
<tr>
<td>ReCLIP</td>
<td>65.48</td>
<td>59.33</td>
<td>52.12</td>
<td>47.87</td>
<td>51.61</td>
<td>50.10</td>
</tr>
<tr>
<td>CLIP Ensemble</td>
<td>25.96</td>
<td>22.32</td>
<td>25.44</td>
<td>23.85</td>
<td>22.00</td>
<td>21.55</td>
</tr>
<tr>
<td>GradCAM</td>
<td>54.00</td>
<td>49.51</td>
<td>48.53</td>
<td>44.96</td>
<td>50.16</td>
<td>48.24</td>
</tr>
<tr>
<td>ReCLIP</td>
<td>65.32</td>
<td>57.70</td>
<td>51.54</td>
<td>47.43</td>
<td>51.80</td>
<td>50.02</td>
</tr>
<tr>
<td>CLIP ViT-B/32</td>
<td>68.08</td>
<td>59.33</td>
<td>52.12</td>
<td>47.87</td>
<td>51.61</td>
<td>50.10</td>
</tr>
</tbody>
</table>

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.

<table>
<thead>
<tr>
<th>Model</th>
<th>Val(_g)</th>
<th>Val(_d)</th>
<th>Test(_A)</th>
<th>Test(_A)</th>
<th>Test(_B)</th>
<th>Test(_B)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random</td>
<td>16.37</td>
<td>15.73</td>
<td>12.45</td>
<td>13.51</td>
<td>21.32</td>
<td>19.20</td>
</tr>
<tr>
<td>UNITER-L (supervised; Chen et al. (2020))</td>
<td>91.84</td>
<td>81.41</td>
<td>92.65</td>
<td>87.04</td>
<td>91.19</td>
<td>74.17</td>
</tr>
<tr>
<td>MDETR (supervised; Kamath et al. (2021))</td>
<td>–</td>
<td>87.51</td>
<td>–</td>
<td>90.40</td>
<td>–</td>
<td>82.67</td>
</tr>
<tr>
<td>Weakly supervised (non-pretrained; Sun et al. (2021))</td>
<td>39.21</td>
<td>38.35</td>
<td>41.14</td>
<td>39.51</td>
<td>37.72</td>
<td>37.01</td>
</tr>
<tr>
<td>CPT-Blk w/ VinVL (Yao et al., 2021)</td>
<td>–</td>
<td>26.9</td>
<td>–</td>
<td>27.5</td>
<td>–</td>
<td>27.4</td>
</tr>
<tr>
<td>CPT-Seg w/ VinVL (Yao et al., 2021)</td>
<td>–</td>
<td>32.2</td>
<td>–</td>
<td>36.1</td>
<td>–</td>
<td>30.3</td>
</tr>
<tr>
<td>CLIP RN50x16</td>
<td>23.31</td>
<td>21.48</td>
<td>19.25</td>
<td>18.56</td>
<td>28.36</td>
<td>25.28</td>
</tr>
<tr>
<td>GradCAM</td>
<td>44.00</td>
<td>40.49</td>
<td>47.41</td>
<td>46.51</td>
<td>43.17</td>
<td>33.66</td>
</tr>
<tr>
<td>ReCLIP w/o relations</td>
<td>40.62</td>
<td>37.61</td>
<td>39.08</td>
<td>38.39</td>
<td>43.55</td>
<td>37.17</td>
</tr>
<tr>
<td>ReCLIP</td>
<td>45.94</td>
<td>45.53</td>
<td>41.24</td>
<td>40.78</td>
<td>52.64</td>
<td>45.55</td>
</tr>
<tr>
<td>CLIP VIT-B/32</td>
<td>25.12</td>
<td>23.79</td>
<td>23.39</td>
<td>22.87</td>
<td>28.42</td>
<td>26.03</td>
</tr>
<tr>
<td>GradCAM</td>
<td>45.41</td>
<td>42.29</td>
<td>50.13</td>
<td>49.04</td>
<td>41.47</td>
<td>36.68</td>
</tr>
<tr>
<td>ReCLIP w/o relations</td>
<td>44.37</td>
<td>40.58</td>
<td>45.09</td>
<td>43.98</td>
<td>43.42</td>
<td>37.63</td>
</tr>
<tr>
<td>ReCLIP</td>
<td>49.69</td>
<td>45.77</td>
<td>48.08</td>
<td>46.99</td>
<td>52.50</td>
<td>45.24</td>
</tr>
<tr>
<td>GradCAM</td>
<td>46.68</td>
<td>42.85</td>
<td>51.99</td>
<td>51.07</td>
<td>40.10</td>
<td>35.21</td>
</tr>
<tr>
<td>ReCLIP w/o relations</td>
<td>45.66</td>
<td>41.97</td>
<td>45.13</td>
<td>43.42</td>
<td>45.40</td>
<td>39.02</td>
</tr>
<tr>
<td>ReCLIP</td>
<td>50.51</td>
<td>45.78</td>
<td>47.11</td>
<td>46.10</td>
<td>54.94</td>
<td>47.07</td>
</tr>
</tbody>
</table>
(a) a man in white shorts and white jacket, walking down a sidewalk.

(b) a man in white jumpsuit with face mask walking.

(c) an african american woman with light colored sweater, brown pants walking down sidewalk near another woman.

(d) woman in blue shirt in doorway.

(e) a man with yellow helmet behind the fence.

(f) a bald black man is walking wearing a tan suit.

(g) a man in all black walking in front of another man.

(h) a man wearing a short-sleeved black top walks by a black car.

(i) a woman in a white top.

(j) a man in a blue polo and brown shorts talking on a cell phone.

Figure 12: Qualitative examples randomly sampled from the RefGTA validation set. Ground-truth referents are in green, MDETR (pre-trained) predictions are in magenta, UNITER (trained on RefCOCO+) predictions are in orange, and ReCLIP predictions are in cyan. The subcaptions are the corresponding referring expressions. For UNITER and ReCLIP, this represents the setting in which we consider all proposals from UniDet.