Deconfounded Visual Grounding

Jianqiang Huang[1,2], **Yu Qin**[2], **Jiaxin Qi**[1], **Qianru Sun**[3], **Hanwang Zhang**[1]
[1]Nanyang Technological University, Singapore
[2]Damo Academy, Alibaba Group
[3]Singapore Management University
jianqiang.jqh@gmail.com, dongdong.qy@alibaba-inc.com, jiaxin003@e.ntu.edu.sg
qianrusun@smu.edu.sg, hanwangzhang@ntu.edu.sg

Abstract

We focus on the confounding bias between language and location in the visual grounding pipeline, where we find that the bias is the major visual reasoning bottleneck. For example, the grounding process is usually a trivial language-location association without visual reasoning, e.g., grounding any language query containing sheep to the nearly central regions, due to that most queries about sheep have ground-truth locations at the image center. First, we frame the visual grounding pipeline into a causal graph, which shows the causalities among image, query, target location and underlying confounder. Through the causal graph, we know how to break the grounding bottleneck: deconfounded visual grounding. Second, to tackle the challenge that the confounder is unobserved in general, we propose a confounder-agnostic approach called: Referring Expression Deconfounder (RED), to remove the confounding bias. Third, we implement RED as a simple language attention, which can be applied in any grounding method. On popular benchmarks, RED improves various state-of-the-art grounding methods by a significant margin. Code is available at: https://github.com/JianqiangH/Deconfounded_VG.

Introduction

Visual Grounding, also known as the task of Referring Expression Comprehension (Karpathy, Joulin, and Fei-Fei 2014; Karpathy and Fei-Fei 2015; Hu et al. 2016), has greatly expanded the application domain of visual detection, from a fixed noun vocabulary, e.g., dog, car, and person, to free-form natural language on demand, e.g., *the dog next to the person who is driving a car*. Regarding the place of candidate proposal generation in the grounding pipeline, there are two traditional camps: two-stage and one-stage. For the two-stage, the first detection stage denotes any object detectors (Ren et al. 2015) for extracting candidate regions from the image, and the second visual reasoning stage is to rank the candidates and select the top one based on their similarities to the query embedding (Yu et al. 2018; Liu et al. 2019a); for the one-stage, the detection and reasoning is unified by directly performing the referent detection: generating the referent region proposals with confidence scores and spatial coordinates for each pixel in the multi-modal feature map, which is usually the concatenation of the visual feature map, location coordinates, and language embeddings (Yang et al. 2019, 2020). However, if the gradients can backpropagate through the visual detector in the two-stage methods, there is little difference between the two camps. Without loss of generality, this paper will be focused on one-stage methods. In particular, the introduction of our approach will be based on the one-stage model (Yang et al. 2019) with YOLO (Bochkovskiy, Wang, and Liao 2020) backbone which has been validated of high inference speed and good generalizability.

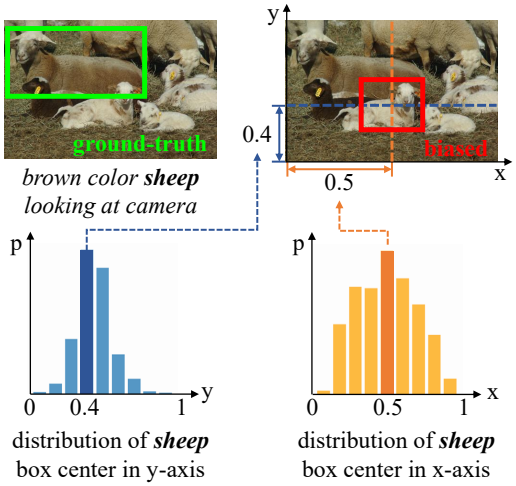

Figure 1: Two image examples: one with ground truth location of *brown color sheep looking at camera*; and the other with a wrong prediction caused by the language bias in the grounding model. Blue bars denote the distribution of the center positions of *sheep* on the y-axis (of the image), and orange bars for the x-axis (of the image).

By analyzing the failure examples of existing grounding methods, we find an ever-overlooked bias, as illustrated in Figure 1: the bounding box prediction tends to be strongly biased to some particular position, e.g., for the subject sheep, the prediction is often on the central area because most of the ground truth locations for sheep related queries are close to the center of the image as shown in the statistics in Figure 1.

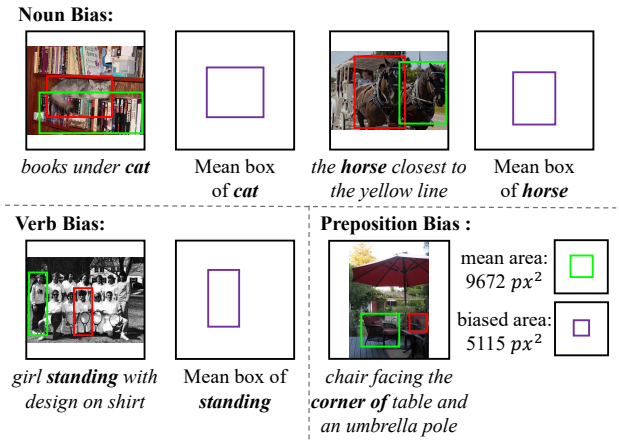

Figure 2: Three types of language bias observed in the grounding model (Yang et al. 2019). The noun and verb biases are from the model trained on RefCOCO+ (Yu et al. 2016) and the preposition bias is from RefCOCOg (Mao et al. 2016). Green boxes denote ground-truth, red boxes denote biased prediction and purple boxes denote language biased location. "Mean box" denotes the average region over all ground truth bounding boxes of a specific language fragment (e.g., horse) in the dataset. For corner of, "mean area" is the average size of all ground truth boxes and "Biased" one is the ground truth boxes averaged over samples with a specific language fragment (e.g., "corner of").

More examples are given in Figure 2, where such bias ubiquitously exists in not only nouns, but also verbs and prepositions. For example, for corner of in Figure 2, the predicted region is biased to be of smaller size (than the size of the ground truth), since most of corner of samples in the dataset are smaller. The model simply takes this bias rather than justifies if the content in this region satisfies the query.

One may realize that the above absurd language bias is due to the *confounder*, which is a common cause of other variables, rendering spurious correlations among them, even if they have no direct causal effects with each other (Pearl, Glymour, and Jewell 2016). For example, the confounding effect of the standing bias in Figure 2 may be due to the common sense that most standing people are the focus of the photographer. Thus, if we fail to consider such an effect that causes the correlation between standing people and the center location in training, when the confounder distribution changes in testing, e.g., most people standing aside, the former correlation in training will be no longer applicable in testing (Pearl and Bareinboim 2014). Readers are encouraged to think about the confounders in other examples. In fact, similar confounding effect is also common in other classical vision tasks such as classification (Yue et al. 2020), detection (Tang, Huang, and Zhang 2020), and segmentation (Zhang et al. 2020). However, compared to those single-modal tasks, the confounder in the visual grounding task is special. As this task is complex and multi-modal, the confounder is unobserved and non-enumerative in general. For example, a general "common sense" is elusive and hard to model.

In this paper, we provide a theoretical ground that addresses why the visual grounding process is confounded by the unobserved confounder, and how we can overcome it. Our technical contributions are three-fold:

- To answer the above question, we frame the visual grounding pipeline into the causal graph. Thanks to the graph, we can model the causalities between language, vision, and locations, and thus offer the causal reasons why the grounding process is confounded.
- As the confounder is unobserved and non-enumerative, we propose a Referring Expression Deconfounder (RED) to estimate the substitute confounder based on the deconfounder theory (Wang and Blei 2019). In this way, we can mitigate the confounding bias without any knowledge on the confounder, i.e., by only using the same observational data just as other baseline methods of visual grounding.
- We implement RED as a simple and effective language attention, whose embedding replaces the corresponding one in any grounding method. Therefore, RED is model-agnostic and thus can help various grounding methods achieve better performances.

Related Work

Visual Grounding In earlier works for visual grounding tasks, most of models are trained in two stages (Plummer et al. 2018; Wang and Specia 2019): its first stage detects region proposals using an off-the-shelf object detector (Ren et al. 2015), and the second stage ranks these regions based on the distances between visual features and language embeddings. Improved methods include using a stronger attention mechanism (Yu et al. 2018; Liu et al. 2019c,b), a better language parsing (Liu et al. 2019a; Niu et al. 2019; Liu et al. 2020) or a better mechanism of filtering out poor region proposals (Chen et al. 2021). While, it is often sub-optimal to train the model in separated stages. Most of recent works (Chen et al. 2018; Liao et al. 2020; Sadhu, Chen, and Nevatia 2019) are thus proposing to exploit a one-stage (i.e., end-to-end) training paradigm. Specifically, they fuse the image features and language embeddings in a dense and optimizable way, and directly predict the grounding result, i.e., the bounding box of the query object. The improved methods (Yang, Li, and Yu 2020; Yang et al. 2020; Shrestha et al. 2020; Huang et al. 2021) are mainly based on incorporating additional head networks that are used in two-stage models.

After careful comparison, we find that regardless of the different types of object detectors (e.g., YOLO and Faster R-CNN), the two-stage and one-stage methods do not have significant difference in terms of the learning pipeline, if all gradients can backpropagate through the backbone of the detector. Without loss of generality, this paper deploys multiple one-stage methods as baselines to show that our method is effective and generic.

Causal Inference. Causal inference (Rubin 2019; Pearl, Glymour, and Jewell 2016; Bareinboim and Pearl 2012) is often used to mitigate spurious correlations. Its methods include deconfounding (Wang et al. 2020; Zhang et al.

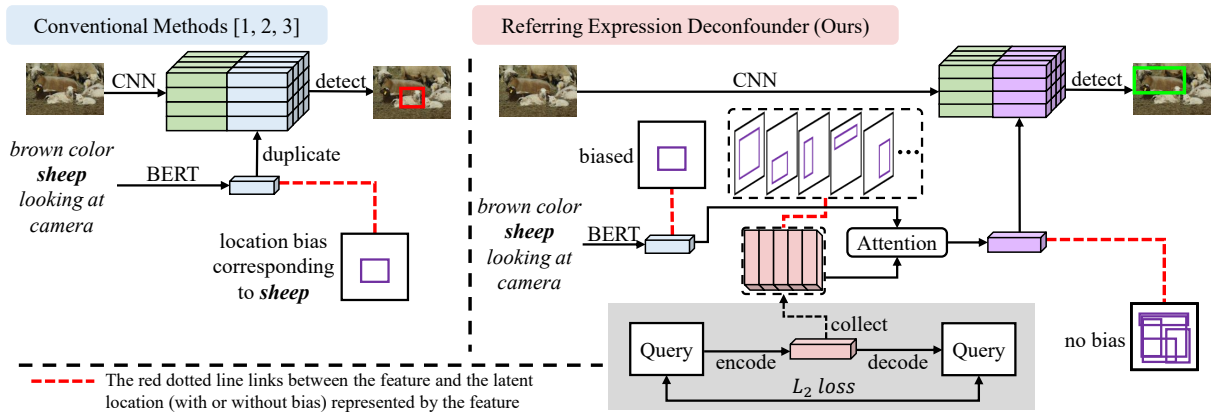

Figure 3: The pipeline of conventional grounding methods vs. our RED. [1, 2, 3] denote the methods in (Yang et al. 2019), (Yang et al. 2020) and (Huang et al. 2021), respectively. In the left part, green bars denote different region features and blue bars denote the duplicated language features. In the right part, the gray region denotes the process of dictionary extraction before training any grounding model. "Query" denotes all the referring expressions in the current training grounding dataset. The dashed black box bounding multiple pink bars represents the dictionary we get by clustering all "pink features" extracted via a trained auto-encoder (in the gray region). The purple bars denote the deconfounded language features derived by our RED.

2020; Yue et al. 2020) and counterfactual inference (Tang, Huang, and Zhang 2020; Tang et al. 2020; Niu et al. 2020). The generic way is to disentangle the factors (in the task), and modeling the existing causal effects (among the factors) on the structural causal graph. In our work, we leverage the method of deconfounding to remove the bias we found in visual grounding datasets. We understand that the confounder is often invisible and non-enumerative from the given dataset. We propose an approach to solve this by generating the proxy of the confounder inspired by a data generation method called Deconfounder (Wang and Blei 2019).

Our approach is different from existing deconfounding methods. For example, (Wang et al. 2020; Yue et al. 2020; Zhang et al. 2020) select a specific substitute for confounder (like object class) and build the confounder set by enumerating its values. (Qi et al. 2020; Yang, Zhang, and Cai 2020b) use a learnable dictionary to learn the values of the confounder in the main training process. In contrast, we build the dictionary by Deconfounder (Wang and Blei 2019) first as an off-the-shelf part, and then in the grounding stage, we directly use the fixed deconfounder set based on which we are able to guarantees the confounder set is the cause of the feature but not vice versa.

Visual Grounding in Causal Graphs

We frame the grounding pipeline summarized in Figure 3 into a structural causal graph (Pearl, Glymour, and Jewell 2016), as shown in Figure 4, where the causalities among the ingredients, e.g., image, language query, and locations, can be formulated. The nodes in the graph denote data variables and the directed links denote the causalities between the nodes. Now, we introduce the graph notations and the rationale behind its construction at a high-level.

Causal Graph

$\boldsymbol{X \rightarrow L \leftarrow R}$. L is an object location. $X \rightarrow L$ and $R \rightarrow L$ denote that L is directly detected from X according to R.
$\boldsymbol{X \leftarrow G \rightarrow R}$. G is the invisible (unobserved) confounder whose distribution is dataset-specific. For example, when G is a dominant visual context (e.g., if there are more samples of horses on grass than horse on road, on grass dominates). $G \rightarrow X$ indicates that most images picturing horse and grass (Zhang et al. 2020), and $G \rightarrow R$ denotes that most language queries R contain the words about horse and grass (Zhang, Niu, and Chang 2018).
$\boldsymbol{G \rightarrow L}$. This is the ever-overlooked causation for the grounding: the confounder G directly causes L, because the ground truth of L is annotated under the same dataset containing the confounder G. For example, under the visual scene horses on grass, $G \rightarrow L$ indicates where and how to put L (e.g., at the middle bottom position, with certain shape and size), which is the root of the bias in the grounding (Zhang et al. 2020).

Causal Inference Process

Given an image, the causal inference process of the grounding task is to maximize the intervention conditional probability of the object location L according to the language query R, which is not equal to the observational conditional probability:

$$P(L \mid do(R), X) \neq P(L \mid R, X), \tag{1}$$

where the do-operation for R denotes that we want to pursue the *causal effect* of R to L by intervening $R = r$ (Pearl, Glymour, and Jewell 2016), i.e., we expect that the model should predict location L that varies from queries R. The inequality is because of the confounding effect (Pearl, Glymour, and Jewell 2016): Shown in Figure 4, even though we conditional on X (i.e., cutting off the backdoor path $R \leftarrow$

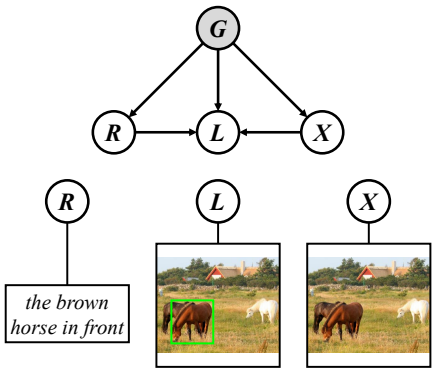

Figure 4: The proposed causal graph and the examples of corresponding nodes. G: unobserved confounder, X: pixel-level image, R: language query, L: the location for the query.

$G \rightarrow X \rightarrow L$), there is still another one, $R \leftarrow G \rightarrow L$, confounding R and L. Therefore, this is the key bottleneck that the grounding pipeline is confounded by language.

Remarks. One may argue that we should also intervene X, because X leads another symmetric backdoor path like R and $P(L|do(R), do(X)) \neq P(L|do(R), X)$. Although this is true in general, we argue that such confounding effect does not only exist in the grounding task, but also ubiquitously exist in any visual recognition tasks (Schölkopf 2019), which is not the main confounding effect in the language-dominant task.

To solve the confounding problem in visual grounding, the general way is to deconfound by using the backdoor adjustment (Pearl, Glymour, and Jewell 2016) through controlling all the values of G:

$$P(L \mid do(R), X) = \mathbb{E}_{g \sim G}[P(L \mid R, X, g)]. \tag{2}$$

Yet, as the confounder is unobserved in general, we cannot enumerate its exact semantic, let alone control it.

Deconfounded Visual Grounding

Deconfounder

Thanks to Deconfounder (Wang and Blei 2019), we can derive a substitute confounder $\hat{G}$ from a data generation model M even if the confounder is unobserved (invisible). Suppose we have the model M that can generate R from $\hat{G}$:

$$P(R \mid \hat{G}) = \prod_{i=1}^{m} P(R_i \mid \hat{G}), \tag{3}$$

where R_i denotes the i-th feature of R (e.g., the i-th dimension or i-th word embedding), respectively. M ensures the independence among the features in R conditional on $\hat{G}$, making $\hat{G}$ include all the confounder stratifications $g \sim G$. Here we give a sketch proof: If a confounder stratification (i.e., a value) g is not included in $\hat{G}$, according to the causal graph illustrated in Figure 4, g can simultaneously cause multiple dimensions in X, R and L, making the different dimensions in R dependent, which contradicts with Eq. (3). Please kindly refer to the formal proof in Appendix.

Therefore, if we have a model M, we can sample $\hat{g} \sim \hat{G}$ to approximate stratifying the original G. By doing this, Eq. (2) can be re-written as:

$$P(L \mid do(R), X) = \mathbb{E}_{\hat{g} \sim \hat{G}}[P(L \mid R, X, \hat{g})]. \tag{4}$$

In practice, to avoid expensive network forward pass calculation, thanks to (Xu et al. 2015; Yang, Zhang, and Cai 2020a), we can apply the Normalized Weighted Geometric Mean to move the outer expectation into the feature level (The proof can be found in Appendix):

$$\begin{aligned} P(L|do(R), X) &= \mathbb{E}_{\hat{g} \sim \hat{G}}[P(L|R, X, \hat{g})] \\ &\approx P(L \mid \mathbb{E}_{\hat{g} \sim \hat{G}}[R, \hat{g}], X). \end{aligned} \tag{5}$$

Next, we detail how to obtain the generative model M and implement the above Eq. (5).

Referring Expression Deconfounder (RED)

Shown in the right part of Figure 3, our proposed approach includes two steps: training the generative model M (illustrated as the shaded part) and training the deconfounded grounding model.

In the first step, we propose to use an auto-encoder to implement the generative model M. The auto-encoder encodes language query R into a hidden embedding, i.e., the substitute confounder $\hat{G}$ (denoted as pink bar), which can be decoded to the query R':

$$\begin{aligned} &\textit{Encoder:} \quad \hat{G} = F_{enc}(R), \\ &\textit{Decoder:} \quad R' = F_{dec}(\hat{G}), \end{aligned} \tag{6}$$

where F_{enc} and F_{dec} are deterministic neural networks, i.e., once R is set to specific values, the values of $\hat{G}$ and R' will be collapsed to a certain value with probability 1. Then, we can use a reconstruction loss $\mathcal{L}_{recon} = d(R, R')$ to train the generative model M, where $d(,)$ is a distance function for the features (e.g., we use Euclidean distance after feature pooling). Note that our deconfounder formulation also supports other non-deterministic generative models such as VAEs (Kingma and Welling 2013; Nielsen et al. 2020) and conditional GANs (Douzas and Bacao 2018).

After we derive the generative model M, we can collect all the samples of substitute confounders $\hat{g} \in \hat{G}$ for R, then cluster them to a dictionary $D_{\hat{g}}$ for $\hat{G}$ (denoted as the stack of pink bars in Figure 3). The reason of clustering is because the number of $\hat{g}$ is too large for backdoor adjustment, while a clustered dictionary can efficiently represent the main elements of $\hat{G}$.

In the second step, we use the dictionary $D_{\hat{g}}$ to perform deconfounded training by Eq.(5). We can instantiate the backdoor adjustment as:

$$\mathbb{E}_{\hat{g} \sim \hat{G}}[R, \hat{g}] = \sum_{\hat{g} \sim D_{\hat{g}}} f(r, \hat{g}) P(\hat{g}), \tag{7}$$

where the right-hand-side is summed over elements in the dictionary $D_{\hat{g}}$, $f(r, \hat{g})$ is a feature fusion function and $P(\hat{g})$ is $1/n$, which is an empirical prior. We implement $f(r, \hat{g})$

Algorithm 1: Visual Grounding with RED

Input: Training images, language queries, and ground-truth locations $\mathcal{D} = \{(x_i, r_i, l_i)\}$

Output: Deconfounded grounding model $P_\theta(L = l \mid do(R = r), X = x)$ in Eq. (9), whose illustration is in Figure 3

Train the generative model F_{enc} and F_{dec} in Eq. (6);
Cluster the substitute confounders to derive $D_{\hat{g}}$;
Initialize grounding model parameters θ randomly;
while not converged **do**
  Extract features (x, r) for sample (x_i, r_i);
  Calculate deconfounded language feature $r' = \sum_{\hat{g}} f(r, \hat{g})P(\hat{g})$ in Eq. (8);
  Calculate deconfounded prediction $P(L = l \mid do(R = r), X = x)$ in Eq. (9);
  Update θ by using the loss in Eq. (10);

as a simple language attention mechanism and derive our deconfounded language feature r' by:

$$\begin{aligned} r' &= \sum\nolimits_{\hat{g} \sim D_{\hat{g}}} f(r, \hat{g}) P(\hat{g}) \\ &= \sum\nolimits_{\hat{g} \sim D_{\hat{g}}} Att(Q = r, K = \hat{g}, V = r) P(\hat{g}), \end{aligned} \tag{8}$$

where $Att(Q, K, V)$ can be any attention (Vaswani et al. 2017) with the Query-Key-Value convention. In this paper, we only adopt a simple top-down attention. Recall in Figure 3 to equip our RED with any conventional grounding method, all we need to do is just to replace the original language embedding feature r (i.e., the light blue bar) with r' (i.e., the purple bar).

Then, the overall deconfounded visual grounding model in Eq. (5) can be implemented as:

$$\begin{aligned} &P(L = l \mid do(R = r), X = x) \\ &\approx P(l \mid (\sum\nolimits_{\hat{g}} f(r, \hat{g}) P(\hat{g})) \oplus x) = P(l \mid r' \oplus x), \end{aligned} \tag{9}$$

where $\oplus$ denotes feature concatenation and l is a location index. In particular, the grounding model $P(l \mid \cdot)$ can be any detection head network that is a softmax layer outputing the probability of the l-th anchor box in the image (Yang et al. 2019, 2020). Specifically, the reason why we only fuse r and $\hat{g}$ without x is because the ignorability principle (Imai and Van Dyk 2004) should be imposed to the variable under causal intervention, that is, R but not X. Table 3 demonstrates that other fusions that take X as input will hurt the performance.

So far, we are ready to train the deconfounded grounding model with the following training losses:

$$\mathcal{L}_{overall} = -\log P(l_{gt} \mid r' \oplus x) + \mathcal{L}_{reg}, \tag{10}$$

where l_{gt} is the ground-truth location and $\mathcal{L}_{reg}$ is a regression loss for the ground-truth bounding box (Yang et al. 2019; Redmon and Farhadi 2018) or mask (Luo et al. 2020). The whole implementation is summarized in Algorithm 1.

Experiments

Datasets

To evaluate RED, we conducted extensive experiments on the following benchmarks. **RefCOCO**, **RefCOCO+** and **RefCOCOg** are three visual grounding benchmarks and their images are from MS-COCO (Lin et al. 2014). Each bounding box from MS-COCO object detection ground truth is annotated with several referring expressions. RefCOCO (Yu et al. 2016) has 50,000 bounding boxes with 142,210 referring expressions in 19,994 images and is split into train/ validation/ testA/ testB with 120,624/ 10,834/ 5,657/ 5,095 images, respectively. RefCOCO+ (Mao et al. 2016) has 19,992 images with 141,564 referring expressions and is split into train/ validation/ testA/ testB with 120,191/ 10,758/ 5,726/ 4,889 images, respectively. Note that, during testing, RefCOCO and RefCOCO+ provide two splits as testA and testB. Images in testA contain multiple people and images in testB contain multiple instances of all other objects. For RefCOCOg, we deploy the popular data partition called RefCOCOg-umd (Nagaraja, Morariu, and Davis 2016), and denote the partitions as val-u and test-u in Table 1. We also conduct experiments on **ReferItGame** (Kazemzadeh et al. 2014) and **Flickr30K Entities** (Plummer et al. 2015). More details can be found in the supplementary materials.

Implementation Details

As our RED is model-agnostic, We followed the same settings of implemented methods to extract the visual representations. As for language representations, before the grounding model training, we used the embedding extracted from uncased version of BERT (Devlin et al. 2018) to train an auto-encoder. Then, we used the trained auto-encoder to extract the substitute confounders for all the training samples R. We deployed the K-Means algorithm to cluster those into $N = 10$ clusters forming the confounder dictionary D_{g^*} in Eq. (7). In the grounding model training stage, we first used a pre-trained frozen BERT to extract language embeddings from the query. Note that we applied the same frozen BERT used in auto-encoder training to make the extracted embeddings consistent with our dictionary. Second, we computed the deconfounded language embeddings by Eq. (8). Then, it will be concatenated to the visual features as shown in Figure 3. Limited by the theory of deconfounder, to ensure the validity of confounder embeddings, we have to use the same structure (i.e., the same frozen BERT structure) to prevent the embeddings gap between the substitute confounders and deconfounded training. As a finetuned BERT is more popular, We will closely follow the breakthroughs of the deconfounder to improve this implementation. More other details of implementations and training settings can be found in the supplementary materials.

Quantitative Results

Comparing with the state-of-the-art (SOTA). In Table 1, we summarize the results of SOTA methods and those *w/* and *w/o* our RED, on RefCOCO-series datasets. We used 4 SOTA methods as our baselines: Yang's-V1 (Yang et al.

		Methods	RefCOCO			RefCOCO+			RefCOCOg	
			val	testA	testB	val	testA	testB	val-u	test-u
Two-Stage	SOTA	MattNet(Yu et al. 2018)	76.40	80.43	69.28	64.93	70.26	56.00	66.67	67.01
		NMT(Liu et al. 2019a)	76.41	81.21	70.09	66.46	72.02	57.52	65.87	66.44
		CM-Att(Liu et al. 2019c)	78.35	83.14	71.32	68.09	73.65	58.03	67.99	68.67
		DGA (Yang et al. 2019)	-	78.42	65.53	-	69.07	51.99	-	63.28
		Ref-NMS (Chen et al. 2021)	80.70	**84.0**	76.04	68.25	73.68	59.42	70.55	70.62
One-Stage	SOTA	RCCF (Liao et al. 2020)	-	81.06	71.85	-	70.35	56.32	-	65.73
		Yang's-V1 (Yang et al. 2019)	72.54	74.35	68.50	56.81	60.23	49.60	61.33	60.36
		Iterative (Sun el al. 2021)	-	74.27	68.10	-	71.05	58.25	-	70.05
		PFOS (Suo et al. 2021)	79.50	81.49	77.13	65.76	69.61	60.30	69.06	68.34
		LBYL (Huang et al. 2021)	79.67	82.91	74.15	68.64	73.38	59.49	-	-
		Yang's V1† (YOLO-V4)	74.67	75.98	70.76	57.21	61.11	50.93	61.89	61.56
		MCN (Luo et al. 2020)	80.08	82.29	74.98	67.16	72.86	57.31	66.46	66.01
		MCN (fixed BERT)	78.93	82.21	73.95	65.54	71.29	56.33	65.76	66.20
	OURS	Yang's-V1 +RED	78.04	79.84	74.58	62.82	66.03	56.72	66.40	66.64
		Yang's-V1†+RED	78.76	80.75	76.19	63.85	66.87	57.51	69.46	69.51
		MCN (fixed BERT) +RED	79.23	83.22	75.23	66.84	72.47	59.03	67.28	67.02
		LBYL +RED	**80.97**	83.20	**77.66**	**69.48**	**73.80**	**62.20**	**71.11**	**70.67**

Table 1: The performance (Acc@0.5%) compared to the state-of-the-art (SOTA) methods on RefCOCO, RefCOCO+ and RefCOCOg, where † denotes applying the visual feature backbone from YOLO-V4, methods with citation denote that the results are from the cited papers. Bold text denotes the best performance. We will use the same notation in the following tables. Note that our reproduced performance of MCN is different from the original paper, because we applied fixed BERT instead of LSTM for language embedding and the reason can be found in "Implementation Details".

Method	ReferIt Game	Flickr30K Entities
ZSGNet (Sadhu el al. 2019)	58.63	63.39
RCCF (Liao et al. 2020)	63.79	-
Yang's-V1 (Yang et al. 2019)	60.67	68.71
Yang's-V2 (Yang et al. 2020)	64.33	69.04
LBYL (Huang et al. 2021)	66.51	-
Yang's-V1† (YOLO-V4)	61.21	70.25
Yang's-V1† +RED	64.37	**70.50**
Yang's-V2 +RED	66.09	69.76
LBYL +RED	**67.27**	-

Table 2: Comparing with SOTA methods on the test sets of ReferItGame and Flickr30K Entities (Acc@0.5%). It is reasonable that ours gets significantly higher gains on the more difficult dataset—ReferIt Game. Please refer to "Comparing with the state-of-the-art (SOTA)".

Methods	RefCOCO+		RefCOCOg	
	testA	testB	val-u	test-u
Yang's-V1†	61.11	50.93	61.89	61.56
+AE($X \oplus R$,$X' \oplus R'$)	66.74	57.23	69.01	68.55
+AE(R,R') (our RED)	**66.87**	**57.51**	**69.46**	**69.51**
+Att(X)+Att(R)	65.57	56.37	67.83	68.11
+Att(R) (our RED)	**66.87**	**57.51**	**69.46**	**69.51**

Table 3: The ablation study of using different auto-encoders and different fusion methods. X and R denote image and language query, respectively. AE(input value, output value) denotes the auto-encoder in Eq. (6) fed with different input values for generating corresponding output values. Att(value) denotes the attention operation in Eq. (8) using different values of V.

2019), Yang's-V2 (Yang et al. 2020), LBYL (Huang et al. 2021), and MCN (Luo et al. 2020). Table 2 shows the results on the ReferItGame and Flickr30K Entities. From an overall view, we observe that using our RED method, which means mitigating the bias in language features, improves the performance of grounding models generally. We elaborate the improvements in the following two points.

RED achieved more improvements for RefCOCOg. This is because RefCOCOg have longer queries than other datasets. Our approach is used to mitigate the bias in language queries. Longer queries contain more bias, so they can benefit more from our approach. For example, our gain on ReferItGame is more than that on Flickr30K Entities, i.e., an increase of 3.16% vs. 0.25%, as most of queries in Flickr30K Entities (e.g., *a man*, *a bench*) are as short as class labels, which contain very little language bias.

In addition, we find that RED can improve more on testB than on testA or val on RefCOCO and RefCOCO+. The reason is that the distribution of testB data is more discrepant than that of training data. It means the testing on testB suffers more from the bias in the grounding model.

RED improved Yang's-V1 (or -V1†) the most. The improvement of RED is brought by applying the deconfounding operation—$do(R)$ (and optionally using $do(X)$). While, those methods with complex co-attention operations and additional visual controlling, e.g., MCN and LBYL, may downplay the effectiveness of deconfounding. In specific, MCN uses extra mask supervision to enforce model looking at the shapes rather than only the bounding boxes. Because

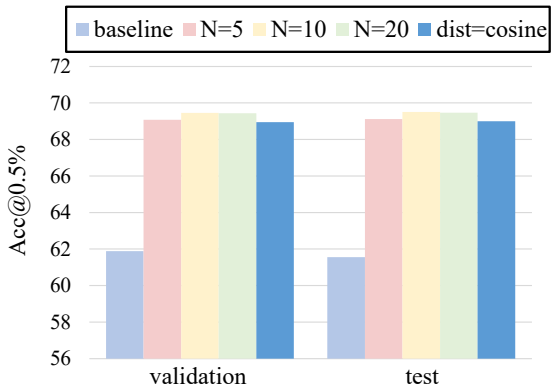

Figure 5: Results using different values of cluster number N and distances for clustering from RefCOCOg. Note that the skyblue, pink, yellow and green columns use Euclidean distance (default), and the blue with $N = 10$ (default).

language confounding effect only influences the location L, the additional shape in the objective function will not be confounded. Therefore, the whole training process of MCN will take less language confounding effects. LBYL splits the image features into four parts combining with queries respectively. This is to intervene X by the value of each part of X, playing the similar role of $do(X)$. Therefore, plugging our RED brings only a partial margin of improvement.

Justification for $do(R)$ and attention for R. We use AE($X \oplus R$,$X' \oplus R'$) to evaluate the performance of language and vision deconfounder for $do(R, X)$, where $\oplus$ denotes concatenation operation. Specifically, before training grounding models, we reconstruct the embeddings of R and X simultaneously, and drive the dictionary by clustering the language and vision confounder using the same way in "**Implementation Details**". As shown in Table 3, we find RED can achieve higher improvements than language and vision deconfounder. The reason may be the embeddings of X need to change during the grounding training, which hurts the consistency of the dictionary of vision deconfounder. Then, we perform attention on both X, R (i.e.,$+Att(X) + Att(R)$) by the substitute confounder, where the attention for X is implemented as a normal channel attention following SENet. We found the performance drop in this ablation. This generally conforms to our statement for Eq (9): To pursue the causal effect from R to L, we need to use the confounder of R and conduct intervention on R.

Hyperparameter selection. We explore different numbers of clustering centers and two clustering distance metrics. As shown in Figure 5, all the clustering numbers N from 5 to 20 show significant improvements over the baseline. After N exceeding 10, the performance won't show further improvement, thus we set $N = 10$. In the comparison of clustering criterion, Euclidean distance shows constant superiority, which is reasonable because of the usage of L_2 loss in the auto-encoder.

Time Consumption. As a significant advantage of the one-stage grounding model is its speed, we test the overhead of our RED in the inference stage to ensure it will not hurt the speed. Under fair settings, we test the speed of Yang's-V1 and Yang's-V1+RED on a single Tesla V100. The results are 24.56ms and 26.53ms per sample, respectively, and the overhead is only 8%, which is reasonable.

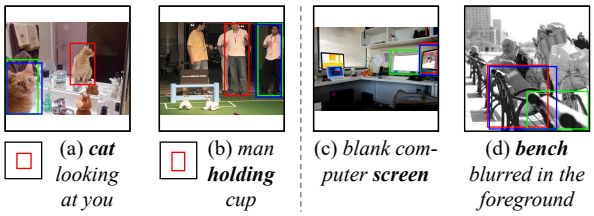

Figure 6: Qualitative examples. The left two examples show that RED mitigates the language bias in the results of Yang's-V1†, where the red boxes denote the biased predictions of Yang's-V1†, blue boxes denote prediction of Yang's-V1†+RED and greens ones are ground-truth boxes. The right two examples show our failure cases.

Qualitative Results

Language Bias Corrections. The qualitative results illustrated in the left half in Figure 6 show that our RED can help models to remove the spurious correlations. In (a) and (b), the location bias is drawn at the left-bottom corner which misleads the grounding models to choose red boxes. After applying our RED, we find that the model can counter such bias (i.e., predict blue boxes near to the ground truth).

Failure Cases. We still find some failure cases shown in the right in Figure 6. In (d), for example, both the red and blue boxes (predicted by baseline and +RED) tend to select the colorful computer screen, which is more common in the dataset. On the contrary, bench is the salient subject compared to the complicated description. That's why both models tend to choose the middle bench.

Conclusions

We investigated the confounding effect that exists ubiquitously in visual grounding models, where the conventional methods may learn spurious correlations between certain language pattern and object location. As the confounder is unobserved in general, we proposed a Referring Expression Deconfounder (RED) approach that seeks a substitute one from observational data. The implementation result of RED is just a simple language attention feature that replaces the language embeddings used in any grounding method. By doing this, RED improves various strong baselines consistently. Our future plans include: 1) We will use other generative models to implement RED (Nielsen et al. 2020). 2) As hacking the unobserved confounder is ill-posed in general (D'Amour 2019), we will try to introduce the large-scale vision-language pre-training priors (Lu et al. 2020, 2019) into the deconfounder.

Acknowledgements

This research is supported, in part, by the Alibaba-NTU Singapore Joint Research Institute (JRI). Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors(s) and do not reflect the views of funding agencies.

References

Bareinboim, E.; and Pearl, J. 2012. Controlling selection bias in causal inference. In *Artificial Intelligence and Statistics*, 100–108.

Bochkovskiy, A.; Wang, C.-Y.; and Liao, H.-Y. M. 2020. YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv preprint arXiv:2004.10934*.

Chen, L.; Ma, W.; Xiao, J.; Zhang, H.; and Chang, S.-F. 2021. Ref-NMS: Breaking Proposal Bottlenecks in Two-Stage Referring Expression Grounding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 1036–1044.

Chen, X.; Ma, L.; Chen, J.; Jie, Z.; Liu, W.; and Luo, J. 2018. Real-time referring expression comprehension by single-stage grounding network. *arXiv preprint arXiv:1812.03426*.

D'Amour, A. 2019. On multi-cause causal inference with unobserved confounding: Counterexamples, impossibility, and alternatives. *arXiv preprint arXiv:1902.10286*.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Douzas, G.; and Bacao, F. 2018. Effective data generation for imbalanced learning using conditional generative adversarial networks. *Expert Systems with applications*, 91: 464–471.

Hu, R.; Xu, H.; Rohrbach, M.; Feng, J.; Saenko, K.; and Darrell, T. 2016. Natural language object retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 4555–4564.

Huang, B.; Lian, D.; Luo, W.; and Gao, S. 2021. Look Before You Leap: Learning Landmark Features for One-Stage Visual Grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 16888–16897.

Imai, K.; and Van Dyk, D. A. 2004. Causal inference with general treatment regimes: Generalizing the propensity score. *Journal of the American Statistical Association*, 99(467): 854–866.

Karpathy, A.; and Fei-Fei, L. 2015. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3128–3137.

Karpathy, A.; Joulin, A.; and Fei-Fei, L. F. 2014. Deep fragment embeddings for bidirectional image sentence mapping. In *Advances in neural information processing systems*, 1889–1897.

Kazemzadeh, S.; Ordonez, V.; Matten, M.; and Berg, T. 2014. Referitgame: Referring to objects in photographs of natural scenes. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 787–798.

Kingma, D. P.; and Welling, M. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Liao, Y.; Liu, S.; Li, G.; Wang, F.; Chen, Y.; Qian, C.; and Li, B. 2020. A Real-Time Cross-modality Correlation Filtering Method for Referring Expression Comprehension. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10880–10889.

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*, 740–755. Springer.

Liu, D.; Zhang, H.; Wu, F.; and Zha, Z.-J. 2019a. Learning to assemble neural module tree networks for visual grounding. In *Proceedings of the IEEE International Conference on Computer Vision*, 4673–4682.

Liu, X.; Li, L.; Wang, S.; Zha, Z.-J.; Meng, D.; and Huang, Q. 2019b. Adaptive reconstruction network for weakly supervised referring expression grounding. In *Proceedings of the IEEE International Conference on Computer Vision*, 2611–2620.

Liu, X.; Wang, Z.; Shao, J.; Wang, X.; and Li, H. 2019c. Improving referring expression grounding with cross-modal attention-guided erasing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1950–1959.

Liu, Y.; Wan, B.; Zhu, X.; and He, X. 2020. Learning Cross-Modal Context Graph for Visual Grounding. In *AAAI*, 11645–11652.

Lu, J.; Batra, D.; Parikh, D.; and Lee, S. 2019. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *Advances in Neural Information Processing Systems*, 13–23.

Lu, J.; Goswami, V.; Rohrbach, M.; Parikh, D.; and Lee, S. 2020. 12-in-1: Multi-task vision and language representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10437–10446.

Luo, G.; Zhou, Y.; Sun, X.; Cao, L.; Wu, C.; Deng, C.; and Ji, R. 2020. Multi-task Collaborative Network for Joint Referring Expression Comprehension and Segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10034–10043.

Mao, J.; Huang, J.; Toshev, A.; Camburu, O.; Yuille, A. L.; and Murphy, K. 2016. Generation and comprehension of unambiguous object descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 11–20.

Nagaraja, V. K.; Morariu, V. I.; and Davis, L. S. 2016. Modeling context between objects for referring expression understanding. In *European Conference on Computer Vision*, 792–807. Springer.

Nielsen, D.; Jaini, P.; Hoogeboom, E.; Winther, O.; and Welling, M. 2020. SurVAE Flows: Surjections to Bridge the Gap between VAEs and Flows. *Advances in Neural Information Processing Systems*, 33.

Niu, Y.; Tang, K.; Zhang, H.; Lu, Z.; Hua, X.-S.; and Wen, J.-R. 2020. Counterfactual VQA: A Cause-Effect Look at Language Bias. *arXiv preprint arXiv:2006.04315*.

Niu, Y.; Zhang, H.; Lu, Z.; and Chang, S.-F. 2019. Variational Context: Exploiting Visual and Textual Context for Grounding Referring Expressions. *IEEE transactions on pattern analysis and machine intelligence*.

Pearl, J.; and Bareinboim, E. 2014. External validity: From do-calculus to transportability across populations. *Statistical Science*, 579–595.

Pearl, J.; Glymour, M.; and Jewell, N. P. 2016. *Causal inference in statistics: A primer*. John Wiley & Sons.

Plummer, B. A.; Kordas, P.; Hadi Kiapour, M.; Zheng, S.; Piramuthu, R.; and Lazebnik, S. 2018. Conditional image-text embedding networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 249–264.

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

Qi, J.; Niu, Y.; Huang, J.; and Zhang, H. 2020. Two causal principles for improving visual dialog. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10860–10869.

Redmon, J.; and Farhadi, A. 2018. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*.

Ren, S.; He, K.; Girshick, R.; and Sun, J. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, 91–99.

Rubin, D. B. 2019. Essential concepts of causal inference: a remarkable history and an intriguing future. *Biostatistics & Epidemiology*, 3(1): 140–155.

Sadhu, A.; Chen, K.; and Nevatia, R. 2019. Zero-shot grounding of objects from natural language queries. In *Proceedings of the IEEE International Conference on Computer Vision*, 4694–4703.

Schölkopf, B. 2019. Causality for machine learning. *arXiv preprint arXiv:1911.10500*.

Shrestha, A.; Pugdeethosapol, K.; Fang, H.; and Qiu, Q. 2020. MAGNet: Multi-Region Attention-Assisted Grounding of Natural Language Queries at Phrase Level. *arXiv preprint arXiv:2006.03776*.

Suo, W.; Sun, M.; Wang, P.; and Wu, Q. 2021. Proposal-free One-stage Referring Expression via Grid-Word Cross-Attention. *arXiv preprint arXiv:2105.02061*.

Tang, K.; Huang, J.; and Zhang, H. 2020. Long-tailed classification by keeping the good and removing the bad momentum causal effect. *Advances in Neural Information Processing Systems*, 33.

Tang, K.; Niu, Y.; Huang, J.; Shi, J.; and Zhang, H. 2020. Unbiased scene graph generation from biased training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3716–3725.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *Advances in neural information processing systems*, 5998–6008.

Wang, J.; and Specia, L. 2019. Phrase Localization Without Paired Training Examples. In *Proceedings of the IEEE International Conference on Computer Vision*, 4663–4672.

Wang, T.; Huang, J.; Zhang, H.; and Sun, Q. 2020. Visual commonsense r-cnn. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10760–10770.

Wang, Y.; and Blei, D. M. 2019. The blessings of multiple causes. *Journal of the American Statistical Association*, 114(528): 1574–1596.

Xu, K.; Ba, J.; Kiros, R.; Cho, K.; Courville, A.; Salakhudinov, R.; Zemel, R.; and Bengio, Y. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, 2048–2057. PMLR.

Yang, S.; Li, G.; and Yu, Y. 2020. Propagating Over Phrase Relations for One-Stage Visual Grounding. In *European Conference on Computer Vision (ECCV)*.

Yang, X.; Zhang, H.; and Cai, J. 2020a. Auto-encoding and Distilling Scene Graphs for Image Captioning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Yang, X.; Zhang, H.; and Cai, J. 2020b. Deconfounded image captioning: A causal retrospect. *arXiv preprint arXiv:2003.03923*.

Yang, Z.; Chen, T.; Wang, L.; and Luo, J. 2020. Improving One-stage Visual Grounding by Recursive Sub-query Construction. *arXiv preprint arXiv:2008.01059*.

Yang, Z.; Gong, B.; Wang, L.; Huang, W.; Yu, D.; and Luo, J. 2019. A fast and accurate one-stage approach to visual grounding. In *Proceedings of the IEEE International Conference on Computer Vision*, 4683–4693.

Yu, L.; Lin, Z.; Shen, X.; Yang, J.; Lu, X.; Bansal, M.; and Berg, T. L. 2018. Mattnet: Modular attention network for referring expression comprehension. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1307–1315.

Yu, L.; Poirson, P.; Yang, S.; Berg, A. C.; and Berg, T. L. 2016. Modeling context in referring expressions. In *European Conference on Computer Vision*, 69–85. Springer.

Yue, Z.; Zhang, H.; Sun, Q.; and Hua, X.-S. 2020. Interventional few-shot learning. *Advances in Neural Information Processing Systems*, 33.

Zhang, D.; Zhang, H.; Tang, J.; Hua, X.-S.; and Sun, Q. 2020. Causal intervention for weakly-supervised semantic segmentation. *Advances in Neural Information Processing Systems*, 33.

Zhang, H.; Niu, Y.; and Chang, S.-F. 2018. Grounding referring expressions in images by variational context. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 4158–4166.