Unbiased Scene Graph Generation from Biased Training

Kaihua Tang[1], Yulei Niu[3], Jianqiang Huang[1,2], Jiaxin Shi[4], Hanwang Zhang[1]
[1]Nanyang Technological University, [2]Damo Academy, Alibaba Group, [3]Renmin University of China, [4]Tsinghua University
kaihua001@e.ntu.edu.sg, niu@ruc.edu.cn, jianqiang.jqh@gmail.com
shijx12@163.com, hanwangzhang@ntu.edu.sg

Abstract

Today's scene graph generation (SGG) task is still far from practical, mainly due to the severe training bias, e.g., collapsing diverse `human walk on/ sit on/lay on beach` *into* `human on beach`*. Given such SGG, the down-stream tasks such as VQA can hardly infer better scene structures than merely a bag of objects. However, debiasing in SGG is not trivial because traditional debiasing methods cannot distinguish between the good and bad bias, e.g., good context prior (e.g.,* `person read book` *rather than* `eat`*) and bad long-tailed bias (e.g.,* `near` *dominating* `behind/in front of`*). In this paper, we present a novel SGG framework based on* ***causal inference*** *but not the conventional likelihood. We first build a causal graph for SGG, and perform traditional biased training with the graph. Then, we propose to draw the* ***counterfactual causality*** *from the trained graph to infer the effect from the bad bias, which should be removed. In particular, we use* ***Total Direct Effect*** *as the proposed final predicate score for unbiased SGG. Note that our framework is agnostic to any SGG model and thus can be widely applied in the community who seeks unbiased predictions. By using the proposed* ***Scene Graph Diagnosis*** *toolkit*[1] *on the SGG benchmark Visual Genome and several prevailing models, we observed significant improvements over the previous state-of-the-art methods.*

1. Introduction

Scene graph generation (SGG) [62] — a visual detection task of objects and their relationships in an image — seems to have never fulfilled its promise: a comprehensive visual scene representation that supports *graph reasoning* for high-level tasks such as visual captioning [67, 65] and VQA [54, 15]. Once equipped with SGG, these high-level tasks have to abandon the ambiguous visual relationships — yet on which are our core efforts made [69, 53, 6], then pretend that there is a graph — nothing but a sparse object layout with binary links, and finally shroud it into graph neural networks [63] for merely more contextual object representations [65, 17, 54]. Although this is partly due to the research gap in graph reasoning [2, 49, 16], the crux lies in the *biased* relationship prediction.

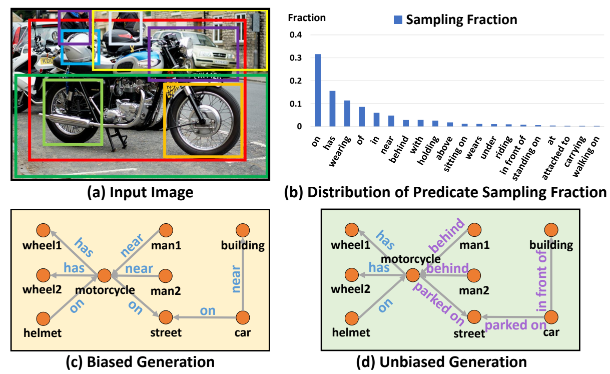

Figure 1. An example of scene graph generation (SGG). (a) An input image with bounding boxes. (b) The distribution of sample fraction for the most frequent 20 predicates in Visual Genome [23]. (c) SGG from re-implemented MOTIFS [69]. (d) SGG by the proposed unbiased prediction from the same model.

Figure 1 visualizes the SGG results from a state-of-the-art model [69]. We can see a frustrating scene: among almost perfectly detected objects, most of their visual relationships are trivial and less informative. For example in Figure 1(c), except the trivial 2D spatial layouts, we know little about the image from `near`, `on`, and `has`. Such heavily biased generation comes from the *biased training data*, more specifically, as shown in Figure 1(b), the highly-skewed long-tailed relationship annotations. For example, if a model is trained for predicting `on` 1,000 times more than `standing on`, then, during test, the former is more likely to prevail over the latter. Therefore, to perform a sensible graph reasoning, we need to distinguish more fine-grained relationships from the ostensibly probable but trivial ones, such as replacing `near` with `behind/in front of`, and `on` with `parking on/driving on` in Figure 1(d).

[1]Our code is publicly available on GitHub: https://github.com/KaihuaTang/Scene-Graph-Benchmark.pytorch

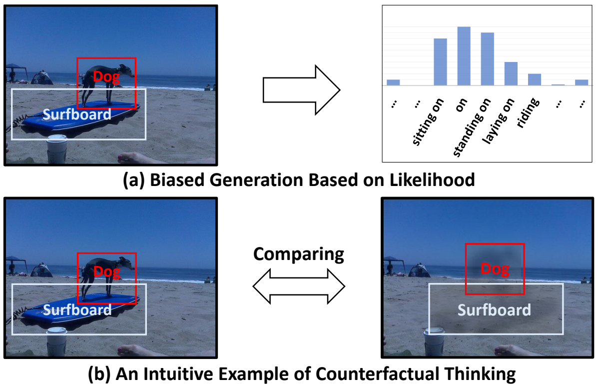

Figure 2. (a) The biased generation that directly predicts labels from likelihood. (b) An intuitive example of the proposed total direct effect, which calculates the difference between the real scene and the counterfactual one. Note that the "wipe-out" is only for the illustrative purpose but not considered as visual processing.

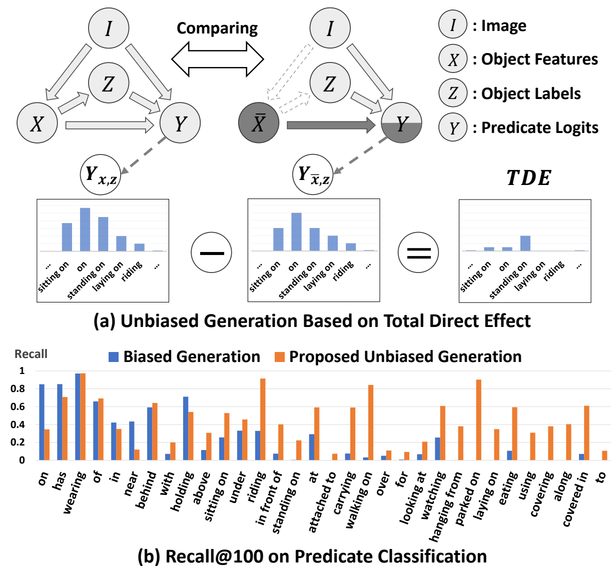

Figure 3. (a) The example of total direct effect calculation and corresponding operations on the causal graph, where $\bar{X}$ represents wiped-out X. (b) Recall@100 of Predicate Classification for selected predicates ranking by sampling fraction. The biased generation refers to re-implemented MOTIFS [69] and the proposed unbiased generation is the result from the same model using TDE.

However, we should not blame the biased training because both our visual world *per se* and the way we describe it are biased: there are indeed more `person carry bag` than `dog carry bag` (*i.e.*, the long-tail theory); it is easier for us to label `person beside table` rather than `eating on` (*i.e.*, bounded rationality [50]); and we prefer to say `person on bike` rather than `person ride on bike` (*i.e.*, language or reporting bias [35]). In fact, most of the biased annotations can help the model learn good contextual prior [32, 69] to filter out the unnecessary search candidates such as `apple park on table` and `apple wear hat`. A promising but embarrassing finding [69] is that: by only using the statistical prior of detected object class in the Visual Genome benchmark [23], we can already achieved 30.1% on Recall@100 for Scene Graph Detection — rendering all the much more complex SGG models almost useless — that is only 1.1-1.5% lower than the state-of-the-art [5, 53, 72]. Not surprisingly, as we will show in Section 5, conventional debiasing methods who do not respect the "good bias" during training, *e.g.*, re-sampling [11] and re-weighting [30], fail to generalize to unseen relationships, *i.e.*, zero-shot SGG [32].

For both machines and humans, decision making is a collaboration of *content* (endogenous reasons) and *context* (exogenous reasons) [56]. Take SGG as an example, in most SGG models [69, 5, 72], the content is the visual features of the subject and object, and the context is the visual features of the subject-object union regions and the pairwise object classes. We humans — born and raised in the biased nature — are ambidextrous in embracing the good while avoiding the bad context, and making unbiased decisions together with the content. The underlying mechanism is *causality-based*: the decision is made by pursuing the main causal effect caused by the content but not the side-effect by context. However, on the other hand, machines are usually *likelihood-based*: the prediction is analogous to look-up the content and its context in a huge likelihood table, interpolated by population training. We believe that the key is to teach machines how to distinguish between the "main effect" and "side-effect".

In this paper, we propose to empower machines the ability of *counterfactual causality* [41] to pursue the "main effect" in unbiased prediction:

> *If I **had not** seen the content, would I still make the same prediction?*

The counterfactual lies between the fact that "I see" and the imagination "I had not", and the comparison between the factual and counterfactual will naturally *remove* the effect from the context bias, because the context is the only thing unchanged between the two alternatives.

To better illustrate the profound yet subtle difference between likelihood and counterfactual causality, we present a `dog standing on surfboard` example in Figure 2(a). Due to the biased training, the model will eventually predict the `on`. Note that even though the rest choices are not all exactly correct, thanks to the bias, they still help to filter out a large amount of unreasonable ones. To take a closer look at what relationship it is in the context bias, we are essentially comparing the original scene with a counterfactual scene (Figure 2(b)): only the visual features of the `dog` and `surfboard` are wiped out, while keeping the rest — the scene and the object classes — untouched, as if the visual features had ever existed. By doing this, we can focus on the main visual effects of the relationship without losing the context.

We propose a novel unbiased SGG method based on the Total Direct Effect (TDE) analysis framework in causal inference [57, 39, 58]. Figure 3(a) shows the underlying causal graphs [40, 41] of the two alternate scenes: factual and counterfactual. Although a formal introduction of them is given in Section 3-4, now you can simply understand the nodes as data features and the directed links as (parametric) data flows. For example, $X \to Y$, $Z \to Y$, and $I \to Y$ indicate that the relationship Y is a combined effect caused by *content*: the pair of object visual features X, *context*: their object classes Z, and *scene*: the image I; the faded links denote that the wiped-out $\bar{X}$ is no longer caused by I or affects Z. These graphs offer an algorithmic formulation to calculate TDE, which exactly realizes the counterfactual thinking in Figure 2. As shown in Figure 3(b), the proposed TDE significantly improves most of the predicates, and impressively, the distribution of the improved performances is no longer long-tailed, indicating the fact that our improvement is indeed from the proposed method, but NOT from the better exploitation of the context bias. A closer analysis in Figure 6 further shows that the worse predictions like `on` — though very few — are due to turning to more fine-grained results such as `stand on` and `park on`. We highlight that TDE is a model-agnostic prediction strategy and thus applicable for a variety of models and fusion tricks [71, 69, 53].

Last but not least, we propose a new standard of SGG diagnosis toolkit (cf. Section 5.2) for more comprehensive SGG evaluations. Besides traditional evaluation tasks, it consists of the bias-sensitive metric: mean Recall [53, 6] and a new Sentence-to-Graph Retrieval for a more comprehensive graph-level metric. By using this toolkit on SGG benchmark Visual Genome [23] and several prevailing baselines, we verify the severe bias in existing models and demonstrate the effectiveness of the proposed unbiased prediction over other debiasing strategies.

2. Related Work

Scene Graph Generation. SGG [62, 69] has received increasing attention in computer vision community, due to the potential revolution that would be brought to down-stream visual reasoning tasks [49, 65, 22, 17]. Most of the existing methods [62, 60, 7, 26, 68, 53, 64, 10, 43, 59] struggle for better feature extraction networks. Zellers *et al.* [69] firstly brought the bias problem of SGG into attention and the followers [53, 6] proposed the unbiased metric (mean Recall), yet, their approaches are still restricted to the feature extraction networks, leaving the biased SGG problem unsolved. The most related work [28] just prunes those dominant and easy-to-predict relationships in the training set.

Unbiased Training. The bias problem has long been investigated in machine learning [55]. Existing debiasing methods can be roughly categorized into three types: 1) data

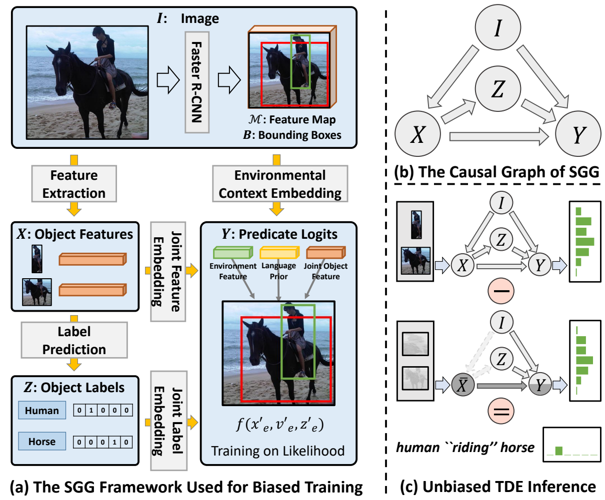

Figure 4. (a) The framework used in our biased training. (b) The causal graph of the SGG framework. (c) An illustration of the proposed TDE inference.

augmentation or re-sampling [9, 25, 27, 11, 3], 2) unbiased learning through elaborately designed training curriculums or learning losses [70, 30], 3) disentangling biased representations from the unbiased [35, 4]. The proposed TDE analysis can be regarded as the third category, but the main difference is that TDE doesn't require to train additional layers like [35, 4] to model the bias, it directly separates the bias from existing models through the counterfactual surgeries on causal graphs.

Mediation Analysis. It is also known as effect analysis [57, 41], which is widely adopted in medical, political or psychological research [45, 19, 8, 33, 21] as the tool of studying the effect of certain treatments or policies. However, it has been neglected in the community of computer vision for years. There are very few recent works [36, 24, 37, 42, 52, 13, 66] trying to endow the model with the capability of causal reasoning. More detailed background knowledge can be found in [40, 41, 57].

3. Biased Training Models in Causal Graph

As illustrated in Figure 4, we summarize the SGG framework in the form of *Causal Graph* (*a.k.a.*, structural causal model) [41, 38, 40]. It is a directed acyclic graph $\mathcal{G} = \{\mathcal{N}, \mathcal{E}\}$, indicating how a set of variables $\mathcal{N}$ interact with each other through the causal links $\mathcal{E}$. It provides a sketch of the causal relations behind the data and how variables obtain their values, *e.g.*, $(I, X, Z) \to Y$. Before we conduct counterfactual analysis that deliberately manipulates the values of nodes and prunes the causal graph, we first revisit the conventional biased SGG model training in the graphical view.

The causal graph in Figure 4(b) is applicable to a variety of SGG methods, since it is highly general, imposing no constraints on the detailed implementations. We casestudy three representative model formulations: the classic VTransE [71], the state-of-the-art MOTIFS [69] and VCTree [53], using the language of nodes and links.

Node I (Input Image&Backbone). A Faster R-CNN [44] is pre-trained and frozen in this node, It outputs a set of bounding boxes $B = \{b_i | i = 1...n\}$ and the feature map $\mathcal{M}$ from image I.

Link $I \rightarrow X$ (Object Feature Extractor). It firstly extracts RoIAlign features [12] $R = \{r_i\}$ and tentative object labels $L = \{l_i\}$ by the object classifier on Faster R-CNN. Then, like MOTIFS [69] or VCTree [53], we can use the following module to encode visual contexts for each object:

$$\textit{Input} : \{(r_i, b_i, l_i)\} \Longrightarrow \textit{Output} : \{x_i\}, \quad (1)$$

where MOTIFS implements it as bidirectional LSTMs (Bi-LSTMs) and VCTree [53] adopts bidirectional TreeLSTMs (Bi-TreeLSTMs) [51], early works like VTransE [71] simply use fully connected layers.

Node X (Object Feature). The pairwise object feature X takes value from $\{(x_i, x_j) | i \neq j; i, j = 1...n\}$. We slightly abuse the notation hereinafter, denoting the combination of representations from i and j as subscript e: $x_e = (x_i, x_j)$.

Link $X \rightarrow Z$ (Object Classification). The fine-tuned label of each object is decoded from the corresponding x_i by:

$$\textit{Input} : \{x_i\} \Longrightarrow \textit{Output} : \{z_i\}, \quad (2)$$

where MOTIFS [69] and VCTree [53] utilizes LSTM and TreeLSTM as decoders to capture the co-occurrence among object labels, respectively. The input of each LSTM/ TreeLSTM cell is the concatenation of feature and the previous label $[x_i; z_{i-1}]$. VTransE [71] uses the conventional fully connected layer as the classifier.

Node Z (Object Class). It contains a pair of one-hot vectors for object labels $z_e = (z_i, z_j)$.

Link $X \rightarrow Y$ (Object Feature Input for SGG). For relationship classification, pairwise feature X are merged into a joint representation by the module:

$$\textit{Input} : \{x_e\} \Longrightarrow \textit{Output} : \{x'_e\}, \quad (3)$$

where another Bi-LSTMs and Bi-TreeLSTMs layers are applied in MOTIFS [69] and VCTree [53], respectively, before concatenating the pair of object features. VTransE [71] uses fully connected layers and element-wise subtraction for feature merging.

Link $Z \rightarrow Y$ (Object Class Input for SGG). The language prior is calculated in this link through a joint embedding layer $z'_e = W_z[z_i \otimes z_j]$, where $\otimes$ generates the one-hot unique vector $\mathbb{R}^{N \times N}$ for the pair of N-way object labels.

Link $I \rightarrow Y$ (Visual Context Input for SGG). This link extracts the contextual union region features $v'_e = \text{Convs}(\text{RoIAlign}(\mathcal{M}, b_i \cup b_j))$ where $b_i \cup b_j$ indicates the union box of two RoIs.

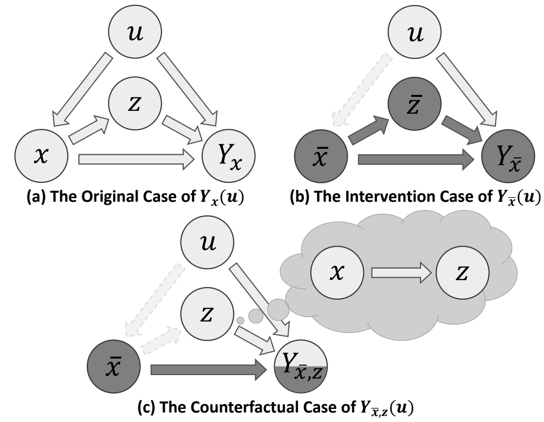

Figure 5. The original causal graph of SGG together with two interventional and counterfactual alternates.

Node Y (Predicate Classification). The final predicate logits Y that takes inputs from the three branches is then generated by using a fusion function. In Section 5, we test two general fusion functions: 1) SUM: $y_e = W_x x'_e + W_v v'_e + z'_e$, 2) GATE: $y_e = W_r x'_e \cdot \sigma(W_x x'_e + W_v v'_e + z'_e)$, where $\cdot$ is element-wise product, $\sigma(\cdot)$ is a sigmoid function.

Training Loss. All models are trained by using the conventional cross-entropy losses of object labels and predicate labels. To avoid any single link spontaneously dominating the generation of logits y_e, especially $Z \rightarrow Y$, we further add auxiliary cross-entropy losses that individually predict y_e from each branch.

4. Unbiased Prediction by Causal Effects

Once the above training has been done, the causal dependencies among the variables are learned, in terms of the model parameters. The conventional biased prediction can only see the output of the entire graph given an image $I = u$ without any idea about how a specific pair of objects affect their predicate. However, causal inference [41] encourages us to think out of the black box. From the graphical point of view, we are no longer required to run the entire graph as a whole. We can directly manipulate the values of several nodes and see what would be going on. For example, we can cut off the link $I \rightarrow X$ and assign a dummy value to X, then investigate what the predicate would be. The above operation is termed *intervention* in causal inference [40]. Next, we will make unbiased predictions by intervention and its induced counterfactuals.

4.1. Notations

Intervention. It can be denoted as $do(\cdot)$. It wipes out all the in-coming links of a variable and demands the variable to take a certain value, *e.g.* $do(X = \bar{x})$ in Figure 5(b), meaning X is no longer affected by its causal parents.

Counterfactual. It means *"counter to the facts"* [46], and takes one step further that assigns the "clash of worlds" combination of values to variables. Take Figure 5(c) as an example, if the intervention $do(X = \bar{x})$ is conducted on X, the variable Z still takes the original z as if x had existed.

Causal Effect. Throughout this section, we will use the pairwise object feature X as our control variable where the intervention is conducted, aiming to assess its effects, due to the fact that there wouldn't be any valid relationship if the pair of objects do not exist. The observed X is denoted as x while the intervened unseen value is $\bar{x}$, which is set to either the mean feature of the training set or zero vector. The object label z on Figure 5(c) is calculated from Eq. (2), taking x as input. We denote the output logits Y after the intervention $X = \bar{x}$ as follows (Figure 5(b)):

$$Y_{\bar{x}}(u) = Y(do(X = \bar{x})|u), \tag{4}$$

where u is the input image in SGG. Following the above notation, the original and counterfactual Y, *i.e.*, Figure 5(a,c), can be re-written as $Y_x(u)$ and $Y_{\bar{x},z}(u)$, respectively.

4.2. Total Direct Effect

As we discussed in Section 1, instead of the static likelihood that tends to be biased, the unbiased prediction lies in the difference between the observed outcome $Y_x(u)$ and its counterfactual alternate $Y_{\bar{x},z}(u)$. The later one is a context-specific bias that we want to remove from prediction. Intuitively, the unbiased prediction that we seek is the visual stimuli from blank to the observed real objects with specific attributes, states, and behaviors, but not merely from the surroundings and language priors. Those specific visual cues of objects are the key to the more fine-grained and informative unbiased predictions, because even if the overall prediction is biased towards the relationship like `dog on surfboard`, the "straight legs" would cause more effect on `standing on` rather than `sitting on`. In causal inference [57, 58], the above prediction process can be calculated as Total Direct Effect (TDE):

$$TDE = Y_x(u) - Y_{\bar{x},z}(u), \tag{5}$$

where the first term is from the original graph and the second one is from the counterfactual, as illustrated in Figure 5.

Note that there is another type of effect [57], Total Effect (TE), which is easy to be mixed up with TDE. Instead of deriving counterfactual bias $Y_{\bar{x},z}(u)$, TE lets all the descendant nodes of X change with intervention $do(X = \bar{x})$ as shown in Figure 5(b). TE is therefore formulated as:

$$TE = Y_x(u) - Y_{\bar{x}}(u). \tag{6}$$

The main difference lies in the fact that $Y_{\bar{x}}(u)$ is not conditioned on the original object labels (those caused by x), so TE only removes the general bias in the whole dataset (similar to the b in $y = k \cdot x + b$), rather than the specific bias caused by the mediator we care about. The subtle difference between TE and TDE is further defined as Natural Indirect Effect (NIE) [57] or Pure Indirect Effect (PIE) [58]. More experimental analyses among these three types of effect are given in Section 5.

Overall SGG. At last, the proposed unbiased prediction $y_e^{\dagger}$ is obtained by replacing the conventional one-time prediction with TDE, which essentially "thinks" twice: one for observational $Y_{x_e}(u) = y_e$, the other for imaginary $Y_{\bar{x},z_e}(u) = y_e(\bar{x}, z_e)$. The unbiased logits of Y is therefore defined as follows:

$$y_e^{\dagger} = y_e - y_e(\bar{x}, z_e). \tag{7}$$

It is also worth mentioning that the proposed TDE doesn't introduce any additional parameters and is widely applicable to a variety of models.

5. Experiments

5.1. Settings and Models

Dataset. For SGG, we used Visual Genome (VG) [23] dataset to train and evaluate our models, which is composed of 108k images across 75k object categories and 37k predicate categories. However, as 92% of the predicates have no more than 10 instances, we followed the widely adopted VG split [62, 69, 53, 5] containing the most frequent 150 object categories and 50 predicate categories. The original split only has training set (70%) and test set (30%). We followed [69] to sample a 5k validation set from training set for parameter tuning. For Sentence-to-Graph Retrieval (cf. Section 5.2), we selected the overlapped 41,859 images between VG and MS-COCO Caption dataset [31] and divided them into train/test-1k/test-5k (35,859/1,000/5,000) sets. The later two only contain images from VG test set in case of exposing to grount-truth SGs. Each image has at least 5 captions serving as human queries, the same as how we use searching engines.

Model Zoo. We evaluated three models: VTransE [71], MOTIFS [69], VTree [53], and two fusion functions: SUM and GATE. They were re-implemented using the same codebase as we proposed. All models shared the same hyper-parameters and the pre-trained detector backbone.

5.2. Scene Graph Generation Diagnosis

Our proposed SGG diagnosis has the following three evaluations:

1. Relationship Retrieval (RR). It can be further divided into three sub-tasks: (1) Predicate Classification **(PredCls)**: taking ground truth bounding boxes and labels as inputs, (2) Scene Graph Classification **(SGCls)**: using ground truth bounding boxes without labels, (3) Scene Graph Detection **(SGDet)**: detecting SGs from scratch. The conventional metric of RR is **Recall@K (R@K)**, which was abandoned in this paper due to the reporting bias [35]. As illustrated in Figure 3(b), previous methods like [69] with good performance on R@K unfairly cater to "head" predicates,

Model	Fusion	Method	Predicate Classification			Scene Graph Classification			Scene Graph Detection		
			mR@20	mR@50	mR100	mR@20	mR50	mR100	mR@20	mR50	mR100
IMP+ [62, 6]	-	-	-	9.8	10.5	-	5.8	6.0	-	3.8	4.8
FREQ [69, 53]	-	-	8.3	13.0	16.0	5.1	7.2	8.5	4.5	6.1	7.1
MOTIFS [69, 53]	-	-	10.8	14.0	15.3	6.3	7.7	8.2	4.2	5.7	6.6
KERN [6]	-	-	-	17.7	19.2	-	9.4	10.0	-	6.4	7.3
VCTree [53]	-	-	14.0	17.9	19.4	8.2	10.1	10.8	5.2	6.9	8.0
MOTIFS†	SUM	Baseline	11.5	14.6	15.8	6.5	8.0	8.5	4.1	5.5	6.8
		Focal	10.9	13.9	15.0	6.3	7.7	8.3	3.9	5.3	6.6
		Reweight	16.0	20.0	21.9	8.4	10.1	10.9	**6.5**	**8.4**	**9.8**
		Resample	14.7	18.5	20.0	9.1	11.0	11.8	5.9	8.2	9.7
		X2Y	13.0	16.4	17.6	6.9	8.6	9.2	5.1	6.9	8.1
		X2Y-Tr	11.6	14.9	16.0	6.5	8.4	9.1	5.0	6.9	8.1
		TE	18.2	25.3	29.0	8.1	12.0	14.0	5.7	8.0	9.6
		NIE	0.6	1.1	1.4	6.1	9.0	10.6	3.8	5.1	6.0
		TDE	**18.5**	**25.5**	**29.1**	**9.8**	**13.1**	**14.9**	5.8	8.2	**9.8**
	GATE	Baseline	12.2	15.5	16.8	7.2	9.0	9.5	5.2	7.2	8.5
		TDE	**18.5**	**24.9**	**28.3**	**11.1**	**13.9**	**15.2**	**6.6**	**8.5**	**9.9**
VTransE†	SUM	Baseline	11.6	14.7	15.8	6.7	8.2	8.7	3.7	5.0	6.0
		TDE	**17.3**	**24.6**	**28.0**	**9.3**	**12.9**	**14.8**	**6.3**	**8.6**	**10.5**
	GATE	Baseline	13.6	17.1	18.6	6.6	8.2	8.7	5.1	6.8	8.0
		TDE	**18.9**	**25.3**	**28.4**	**9.8**	**13.1**	**14.7**	**6.0**	**8.5**	**10.2**
VCTree†	SUM	Baseline	11.7	14.9	16.1	6.2	7.5	7.9	4.2	5.7	6.9
		TDE	**18.4**	**25.4**	**28.7**	**8.9**	**12.2**	**14.0**	**6.9**	**9.3**	**11.1**
	GATE	Baseline	12.4	15.4	16.6	6.3	7.5	8.0	4.9	6.6	7.7
		TDE	**17.2**	**23.3**	**26.6**	**8.9**	**11.8**	**13.4**	**6.3**	**8.6**	**10.3**

Table 1. The SGG performances of Relationship Retrieval on mean Recall@K [53, 6]. The SGG models re-implemented under our codebase are denoted by the superscript †.

e.g., `on`, while neglect the "tail" ones, *e.g.*, predicates like `parked on`, `laying on` have embarrassingly 0.0 Recall@100. To speak for the valuable "tail" rather than the trivial "head", we adopted a recent replacement, **mean Recall@K (mR@K)**, proposed by Tang *et al*. [53] and Chen *et al*. [6]. mR@K retrieves each predicate separately and then averages R@K for all predicates.

2. Zero-Shot Relationship Retrieval (ZSRR). It was introduced by Lu *et al*. [32] as **Zero-Shot Recall@K** and was firstly evaluated on VG dataset in this paper, which only reports the R@K of those subject-predicate-object triplets that have never been observed in the training set. ZSRR also has three sub-tasks as RR.

3. Sentence-to-Graph Retrieval (S2GR). It uses the image caption sentence as the query to retrieve images represented as SGs. Both RR and ZSRR are triplet-level evaluations, ignoring the graph-level coherence. Therefore, we design S2GR, using human descriptions to retrieve detected SGs. We didn't use proxy vision-language tasks like captioning [65, 67] and VQA [54, 15] as the diagnosis, because their implementations have too many components unrelated to SGG and their datasets are challenged by their own biases [1, 14, 34]. In S2GR, the detected SGs (using SGDet) are regarded as the only representations of images, cut off all the dependencies on black-box visual features, so any bias on SGG would sensitively violate the coherence of SGs, resulting in worse retrieval results. For example, if `walking on` was detected as the biased alternative `on`, images would be mixed up with those have `sitting on` or `laying on`. Note that S2GR is fundamentally different

Model	Fusion	Method	PredCls	SGCls	SGDet
Zero-Shot Relationship Retrieval			R@50/100	R@50/100	R@50/100
MOTIFS†	SUM	Baseline	10.9 / 14.5	2.2 / 3.0	0.1 / 0.2
		Focal	10.9 / 14.4	2.2 / 3.1	0.1 / 0.3
		Reweight	0.7 / 0.9	0.1 /0.1	0.0 / 0.0
		Resample	11.1 / 14.3	2.3 / 3.1	0.1 / 0.3
		X2Y	11.8 / 17.6	2.3 / 3.7	1.6 / 2.7
		X2Y-Tr	13.7 / 17.6	3.1 / 4.2	1.8 / 2.8
		TE	14.2 / 18.1	1.4 / 2.0	1.4 / 1.8
		NIE	2.4 / 3.2	0.2 / 0.4	0.3 / 0.6
		TDE	**14.4 / 18.2**	**3.4 / 4.5**	**2.3 / 2.9**
	GATE	Baseline	7.4 / 10.6	0.9 / 1.3	0.2 / 0.4
		TDE	**7.7 / 11.0**	**1.9 / 2.6**	**1.9 / 2.5**
VTransE†	SUM	Baseline	11.3 / 14.7	2.5 / 3.3	0.8 / 1.5
		TDE	**13.3 / 17.6**	**2.9 / 3.8**	**2.0 / 2.7**
	GATE	Baseline	4.2 / 5.9	1.9 / 2.6	**1.9** / 2.6
		TDE	**5.3 / 7.9**	**2.1 / 3.0**	**1.9 / 2.7**
VCTree†	SUM	Baseline	10.8 / 14.3	1.9 / 2.6	0.2 / 0.7
		TDE	**14.3 / 17.6**	**3.2 / 4.0**	**2.6 / 3.2**
	GATE	Baseline	4.4 / 6.8	2.5 / 3.3	1.8 / 2.7
		TDE	**5.9 / 8.1**	**3.0 / 3.7**	**2.2 / 2.8**

Table 2. The results of Zero-Shot Relationship Retrieval.

from the previous image retrieval with scene graph [18, 48], because the latter still consider the images as visual features but not SGs. **Recall@20/100 (R@20/100)** and median ranking indexes of retrieved results **(Med)** on the gallery size of 1,000 and 5,000 were evaluated. Note that S2GR should have diverse implementations as long as its spirit: graph-level symbolic retrieval, is fulfilled. We provide our implementation in the next sub-section.

5.3. Implementation Details

Object Detector. Following the previous works [62, 69, 53], we pre-trained a Faster R-CNN [44] and froze it to be the underlying detector of our SGG models. We equipped the Faster R-CNN with a ResNeXt-101-FPN [29, 61] back-

Sentence-to-Graph Retrieval								
Gallery Size			1000			5000		
Model	Fusion	Method	R@20	R@100	Med	R@20	R@100	Med
MOTIFS†	SUM	Baseline	11.6	39.9	155	3.1	12.1	708
		Focal	10.9	39.0	163	2.9	11.1	737
		Reweight	9.7	36.8	159	3.0	11.4	725
		Resample	13.1	43.6	124	2.5	13.4	593
		X2Y	14.3	44.8	125	3.5	14.6	556
		X2Y-Tr	14.5	45.6	114	3.9	16.8	525
		TE	15.9	49.9	100	4.4	16.9	469
		NIE	6.7	29.2	202	1.6	8.6	1050
		TDE	**17.0**	**53.6**	**91**	**5.2**	**18.9**	**425**
	GATE	Baseline	13.7	45.6	143	4.4	16.2	618
		TDE	**20.8**	**59.2**	**72**	**5.2**	**21.3**	**325**
VTransE†	SUM	Baseline	12.3	42.3	129	**3.6**	15.0	596
		TDE	**14.7**	**48.4**	**106**	**3.6**	**16.3**	**483**
	GATE	Baseline	12.9	41.8	136	3.8	14.3	634
		TDE	**18.5**	**50.4**	**110**	**4.5**	**19.1**	**486**
VCTree†	SUM	Baseline	9.9	37.4	150	3.1	11.5	745
		TDE	**19.0**	**57.0**	**82**	**5.0**	**20.0**	**385**
	GATE	Baseline	13.4	44.1	121	3.7	13.6	583
		TDE	**19.1**	**55.5**	**87**	**5.1**	**20.3**	**395**

Table 3. The results of Sentence-to-Graph Retrieval.

bone and scaled the longer side of input images to be 1k pixels. The detector was trained on the training set of VG using SGD as optimizer. We set the batch size to 8 and the initial learning rate to 8×10^{-3}, which was decayed by the factor of 10 on the 30k$^{\text{th}}$ and 40k$^{\text{th}}$ iterations. The final detector achieved 28.14 mAP on VG test set (using 0.5 IoU threshold). 4 2080ti GPUs were used for the pre-training.

Scene Graph Generation. On top of the frozen detector, we trained SGG models using SGD as optimizer. Batch size and initial learning rate were set to be 12 and 12×10^{-2} for PredCls and SGCls; 8 and 8×10^{-2} for SGDet. The learning rate would be decayed by 10 two times after the validation performance plateaus. For SGDet, 80 RoIs were sampled for each image and Per-Class NMS [47, 69] with 0.5 IoU was applied in object prediction. We sampled up to 1,024 subject-object pairs containing 75% background pairs during training. Different from previous works [69, 53, 5], we didn't assume that non-overlapping subject-object pairs are invalid in SGDet, making SGG more general.

Sentence-to-Graph Retrieval. We handled S2GR as a graph-to-graph matching problem. The query captions of each image were stuck together and parsed to a text-SG using [48]. We set all the subject/object and predicates that appear less than 5 times to "UNKNOWN" tokens, obtaining a dictionary of size 4,459 subject/object entities and 645 predicates, respectively. The original image SG generated from SGDet contains a fixed number of RoIs and forces all valid subject-object pairs to predict foreground relationships, to serve the K number in mR@K, which is inappropriate for S2GR. Therefore, we used a threshold of 0.1 to filter RoIs by the label probabilities and removed all background predicates from the graph. Recall that the vocabulary size of the entity and predicate for image SGs are 150 and 50 as we mentioned before. To match the two heterogeneous graphs: image SG and text SG, in a unified space, we used BAN [20] to encode the two graph types into fixed-dimension vectors to facilitate the retrieval. More details can be found in supplementary material.

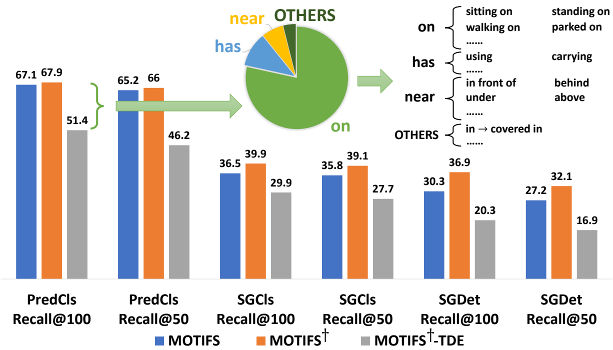

Figure 6. The pie chart summarizes all the relationships, that are correctly detected by the baseline model but considered "incorrect" by TDE. The right side of the pie chart shows the corresponding labels given by the TDE. Combining with our qualitative examples, we believe that the drop of Recall@K is caused by two reasons: 1) the annotators preference towards simple annotations caused by bounded rationality [50], 2) TDE tends to predict more action-like relationships rather than vague prepositions.

5.4. Ablation Studies

Except for the models and fusion functions that we've discussed before, we also investigated three conventional debiasing methods, two intuitive causal graph surgeries, and other two types of causal effects: 1) **Focal**: focal loss [30] automatically penalizes well-learned samples and focuses on the hard ones. We followed the hyper-parameters ($\gamma = 2.0, \alpha = 0.25$) optimized in [30]. 2) **Reweight**: weighted cross-entropy is widely used in the industry for biased data. The inversed sample fractions were assigned to each predicate category as weights. 3) **Resample** [3]: rare categories were up-sampled by the inversed sample fraction during training. 4) **X2Y**: since we argued that the unbiased effect was under the effect of object features X, it directly generated SG by the outputs of $X \rightarrow Y$ branch after biased training. 5) **X2Y-Tr**: it even cut off other branches, using $X \rightarrow Y$ for both training and testing. 6) **TE**: as we introduced in Section 4, TE is the debiasing method that not conditioned on the contexts. 7) **NIE**: it is the marginal difference between TDE and TE, *i.e.*, NIE = TE-TDE, which can be considered as the pure effect caused by introducing the bias $Z \rightarrow Y$. **NOTE**: although zero vector can also be used as the wiped-out input $\bar{x}$, we chose the mean feature of training set for minor improvements.

5.5. Quantitative Studies

RR & ZSRR. The results are listed in Table 1& 2. Despite the fact that conventional debiasing methods: Reweight and Resample, directly hack the mR@K metric, they only gained limited advantages in RR but not in ZSRR. In contrast to the high mR@K of Reweight in RR SGDet, it got embarrassingly $0.0/0.0$ in ZSRR SGDet, indicating that such debiased training methods ruin the useful context prior. Focal loss [30] barely worked for both RR and ZSRR.

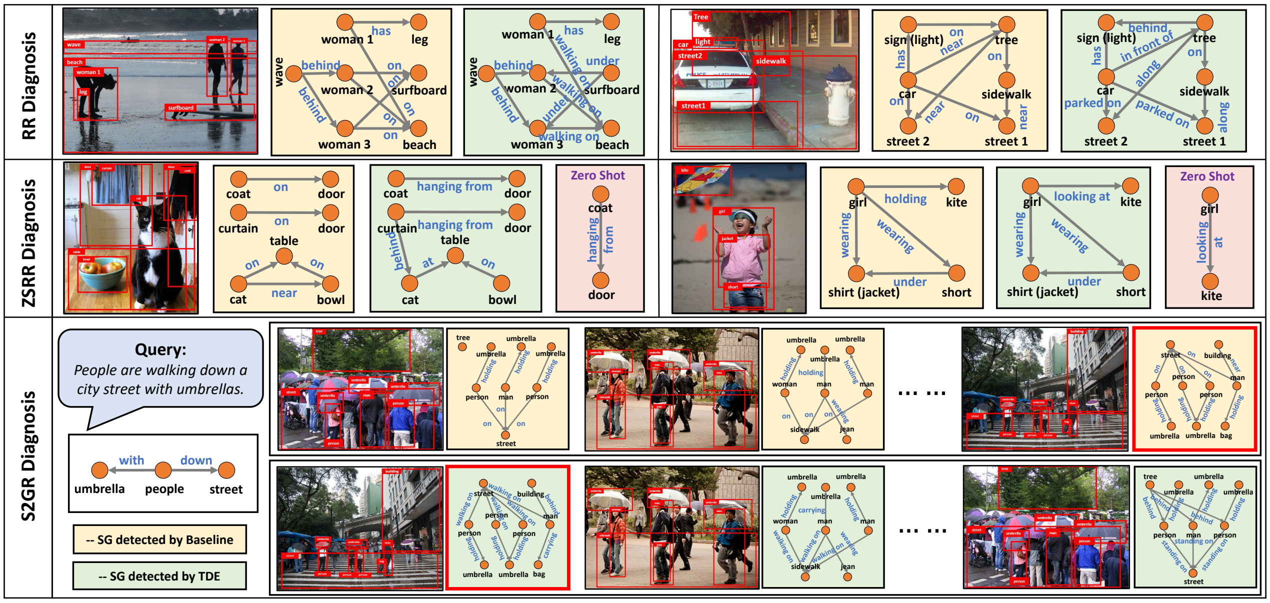

Figure 7. Results of scene graphs generated from MOTIF†-SUM baseline (yellow) and corresponding TDE (green). Top: relationship retrieval results. Mid: zero shot relationship retrieval results. Red boxes indicate the zero shot triplets. Bottom: results of S2GR. Red boxes mean the correctly retrieved SGs. Part of the trivial detected objects are removed from the graphs, due to space limitation.

Causal graph surgeries, X2Y and X2Y-Tr, both improved RR and ZSRR from the baseline, yet their increases were limited. TE had a very similar performance to TDE, but as we discussed, it removed the general bias rather than the subject-object specific bias. NIE is the marginal improvements from TE to TDE, which was even worse than baseline. Although R@K is not a qualified metric for RR as we discussed, we still reported the R@50/100 performance of MOTIFS†-SUM in Figure 6. We can observe a performance drop from baseline to TDE, but a further analysis shows that those considered as correct in baseline and "incorrect" in TDE were mainly the "head" predicates, and they are classified by TDE into more fine-grained "tail" classes. Among all three models and two fusion functions, even the worst TDE performance outperforms previous state-of-the-art methods [53, 6] by a large margin on RR mR@K.

S2GR. In S2GR, Focal and Reweight are even worse than the baseline. Among all the three conventional debiasing methods, Resample was the most stable one based on our experiments. X2Y and X2Y-Tr have minor advantages over baseline. TE takes the 2nd place and was only a little bit worse than TDE. NIE is the worst as we expected because it is only based on the pure context bias. It is worth highlighting that all the three models and two fusion functions had significant improvements after we applied TDE.

5.6. Qualitative Studies

We visualized several SGCls examples that generated from MOTIFS†-SUM baseline and TDE in the top and mid rows of Figure 7, scene graphs generated by TDE are much more discriminative compared to the baseline model which prefers trivial predicates like `on`. The right half of the mid row shows that the baseline model would even generate `holding` due to the long-tail bias when the girl is not touching the kite, implying that the biased predictions are easy to be "blind", while TDE successfully predicted `looking at`. The bottom of Figure 7 is an example of S2GR, where the SGs detected by baseline model lost the detailed actions of people, considering both `person walking on street` and `person standing on street` as `person on street`, which caused worse retrieval results. All the examples show a clear trend that TDE is much more sensitive to those semantically informative relationships instead of the trivially biased ones.

6. Conclusions

We presented a general framework for unbiased SGG from biased training, and this is the first work addressing the serious bias issue in SGG. With the power of *counterfactual causality*, we can remove the harmful bias from the good context bias, which cannot be easily identified by traditional debiasing methods such as data augmentation [9, 11] and unbiased learning [30]. We achieved the unbiasedness by calculating the Total Direct Effect (TDE) with the help of a causal graph, which is a roadmap for training any SGG model. By using the proposed Scene Graph Diagnosis toolkit, our unbiased SGG results are considerably better than their biased counterparts.

Acknowledgments We'd like to thank all reviewers for their constructive comments. This work was partially supported by the NTU-Alibaba JRI.

References

[1] A. Agrawal, D. Batra, D. Parikh, and A. Kembhavi. Don't just assume; look and answer: Overcoming priors for visual question answering. In *CVPR*, 2018.

[2] P. W. Battaglia, J. B. Hamrick, V. Bapst, A. Sanchez-Gonzalez, V. Zambaldi, M. Malinowski, A. Tacchetti, D. Raposo, A. Santoro, R. Faulkner, et al. Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261*, 2018.

[3] E. Burnaev, P. Erofeev, and A. Papanov. Influence of resampling on accuracy of imbalanced classification. In *ICMV*, 2015.

[4] R. Cadene, C. Dancette, H. Ben-younes, M. Cord, and D. Parikh. Rubi: Reducing unimodal biases in visual question answering. *arXiv preprint arXiv:1906.10169*, 2019.

[5] L. Chen, H. Zhang, J. Xiao, X. He, S. Pu, and S.-F. Chang. Counterfactual critic multi-agent training for scene graph generation. In *ICCV*, 2019.

[6] T. Chen, W. Yu, R. Chen, and L. Lin. Knowledge-embedded routing network for scene graph generation. In *CVPR*, 2019.

[7] B. Dai, Y. Zhang, and D. Lin. Detecting visual relationships with deep relational networks. In *CVPR*, 2017.

[8] G. Dunn, R. Emsley, H. Liu, S. Landau, J. Green, I. White, and A. Pickles. Evaluation and validation of social and psychological markers in randomised trials of complex interventions in mental health: a methodological research programme. *NIHR Journals Library*, 2015.

[9] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. A. Wichmann, and W. Brendel. Imagenet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. In *International Conference on Learning Representations*, 2019.

[10] J. Gu, H. Zhao, Z. Lin, S. Li, J. Cai, and M. Ling. Scene graph generation with external knowledge and image reconstruction. In *CVPR*, 2019.

[11] H. He and E. A. Garcia. Learning from imbalanced data. *IEEE Transactions on knowledge and data engineering*, 2009.

[12] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *ICCV*, 2017.

[13] Y. He, Y. Ding, P. Liu, L. Zhu, H. Zhang, and Y. Yang. Learning filter pruning criteria for deep convolutional neural networks acceleration. In *CVPR*, 2020.

[14] L. A. Hendricks, K. Burns, K. Saenko, T. Darrell, and A. Rohrbach. Women also snowboard: Overcoming bias in captioning models. In *ECCV*. Springer, 2018.

[15] D. A. Hudson and C. D. Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *CVPR*, 2019.

[16] D. A. Hudson and C. D. Manning. Learning by abstraction: The neural state machine. *NeurIPS*, 2019.

[17] J. Johnson, A. Gupta, and L. Fei-Fei. Image generation from scene graphs. In *CVPR*, 2018.

[18] J. Johnson, R. Krishna, M. Stark, L.-J. Li, D. Shamma, M. Bernstein, and L. Fei-Fei. Image retrieval using scene graphs. In *CVPR*, 2015.

[19] L. Keele. The statistics of causal inference: A view from political methodology. *Political Analysis*, 2015.

[20] J.-H. Kim, J. Jun, and B.-T. Zhang. Bilinear attention networks. In *Advances in Neural Information Processing Systems*, 2018.

[21] B. G. King. A political mediation model of corporate response to social movement activism. *Administrative Science Quarterly*, 2008.

[22] R. Krishna, I. Chami, M. Bernstein, and L. Fei-Fei. Referring relationships. In *CVPR*, 2018.

[23] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 2017.

[24] M. J. Kusner, J. Loftus, C. Russell, and R. Silva. Counterfactual fairness. In *Advances in Neural Information Processing Systems*, 2017.

[25] Y. Li, Y. Li, and N. Vasconcelos. Resound: Towards action recognition without representation bias. In *ECCV*, 2018.

[26] Y. Li, W. Ouyang, B. Zhou, K. Wang, and X. Wang. Scene graph generation from objects, phrases and caption regions. In *ICCV*, 2017.

[27] Y. Li and N. Vasconcelos. Repair: Removing representation bias by dataset resampling. In *CVPR*, 2019.

[28] Y. Liang, Y. Bai, W. Zhang, X. Qian, L. Zhu, and T. Mei. Vrr-vg: Refocusing visually-relevant relationships. In *ICCV*, pages 10403–10412, 2019.

[29] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017.

[30] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár. Focal loss for dense object detection. In *ICCV*, 2017.

[31] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *ECCV*. Springer, 2014.

[32] C. Lu, R. Krishna, M. Bernstein, and L. Fei-Fei. Visual relationship detection with language priors. In *ECCV*, 2016.

[33] D. P. MacKinnon, A. J. Fairchild, and M. S. Fritz. Mediation analysis. *Annu. Rev. Psychol.*, 2007.

[34] V. Manjunatha, N. Saini, and L. S. Davis. Explicit bias discovery in visual question answering models. In *CVPR*, 2019.

[35] I. Misra, C. Lawrence Zitnick, M. Mitchell, and R. Girshick. Seeing through the human reporting bias: Visual classifiers from noisy human-centric labels. In *CVPR*, 2016.

[36] S. Nair, Y. Zhu, S. Savarese, and L. Fei-Fei. Causal induction from visual observations for goal directed tasks. *arXiv preprint arXiv:1910.01751*, 2019.

[37] Y. Niu, K. Tang, H. Zhang, Z. Lu, X. Hua, and J.-R. Wen. Counterfactual vqa: A cause-effect look at language bias. *arXiv*, 2020.

[38] J. Pearl. *Causality: models, reasoning and inference*. Springer, 2000.

[39] J. Pearl. Direct and indirect effects. In *Proceedings of the 17th conference on uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., 2001.

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

[41] J. Pearl and D. Mackenzie. *THE BOOK OF WHY: THE NEW SCIENCE OF CAUSE AND EFFECT*. Basic Books, 2018.

[42] J. Qi, Y. Niu, J. Huang, and H. Zhang. Two causal principles for improving visual dialog. *arXiv preprint arXiv:1911.10496*, 2019.

[43] M. Qi, W. Li, Z. Yang, Y. Wang, and J. Luo. Attentive relational networks for mapping images to scene graphs. In *CVPR*, 2019.

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

[45] L. Richiardi, R. Bellocco, and D. Zugna. Mediation analysis in epidemiology: methods, interpretation and bias. *International journal of epidemiology*, 2013.

[46] N. J. Roese. Counterfactual thinking. *Psychological bulletin*, 1997.

[47] A. Rosenfeld and M. Thurston. Edge and curve detection for visual scene analysis. *IEEE Transactions on computers*, 1971.

[48] S. Schuster, R. Krishna, A. Chang, L. Fei-Fei, and C. D. Manning. Generating semantically precise scene graphs from textual descriptions for improved image retrieval. In *Proceedings of the fourth workshop on vision and language*, 2015.

[49] J. Shi, H. Zhang, and J. Li. Explainable and explicit visual reasoning over scene graphs. In *CVPR*, 2019.

[50] H. A. Simon. Bounded rationality. In *Utility and probability*. Springer, 1990.

[51] K. S. Tai, R. Socher, and C. D. Manning. Improved semantic representations from tree-structured long short-term memory networks. In *ACL*, 2015.

[52] H. Z. Q. S. Tan Wang, Jianqiang Huang. Visual commonsense r-cnn. In *Conference on Computer Vision and Pattern Recognition*, 2020.

[53] K. Tang, H. Zhang, B. Wu, W. Luo, and W. Liu. Learning to compose dynamic tree structures for visual contexts. In *CVPR*, 2019.

[54] D. Teney, L. Liu, and A. van den Hengel. Graph-structured representations for visual question answering. In *CVPR*, 2017.

[55] A. Torralba, A. A. Efros, et al. Unbiased look at dataset bias. In *CVPR*, 2011.

[56] N. Van Hoeck, P. D. Watson, and A. K. Barbey. Cognitive neuroscience of human counterfactual reasoning. *Frontiers in human neuroscience*, 2015.

[57] T. VanderWeele. *Explanation in causal inference: methods for mediation and interaction*. Oxford University Press, 2015.

[58] T. J. VanderWeele. A three-way decomposition of a total effect into direct, indirect, and interactive effects. *Epidemiology (Cambridge, Mass.)*, 2013.

[59] W. Wang, R. Wang, S. Shan, and X. Chen. Exploring context and visual pattern of relationship for scene graph generation. In *CVPR*, 2019.

[60] S. Woo, D. Kim, D. Cho, and I. S. Kweon. Linknet: Relational embedding for scene graph. In *Advances in Neural Information Processing Systems*, 2018.

[61] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He. Aggregated residual transformations for deep neural networks. In *CVPR*, 2017.

[62] D. Xu, Y. Zhu, C. B. Choy, and L. Fei-Fei. Scene graph generation by iterative message passing. In *CVPR*, 2017.

[63] S. Yan, Y. Xiong, and D. Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *AAAI*, 2018.

[64] J. Yang, J. Lu, S. Lee, D. Batra, and D. Parikh. Graph r-cnn for scene graph generation. In *ECCV*, 2018.

[65] X. Yang, K. Tang, H. Zhang, and J. Cai. Auto-encoding scene graphs for image captioning. In *CVPR*, 2019.

[66] X. Yang, H. Zhang, and J. Cai. Deconfounded image captioning: A causal retrospect, 2020.

[67] T. Yao, Y. Pan, Y. Li, and T. Mei. Exploring visual relationship for image captioning. In *ECCV*, 2018.

[68] G. Yin, L. Sheng, B. Liu, N. Yu, X. Wang, J. Shao, and C. Change Loy. Zoom-net: Mining deep feature interactions for visual relationship recognition. In *ECCV*, 2018.

[69] R. Zellers, M. Yatskar, S. Thomson, and Y. Choi. Neural motifs: Scene graph parsing with global context. In *CVPR*, 2018.

[70] R. Zemel, Y. Wu, K. Swersky, T. Pitassi, and C. Dwork. Learning fair representations. In *ICML*, 2013.

[71] H. Zhang, Z. Kyaw, S.-F. Chang, and T.-S. Chua. Visual translation embedding network for visual relation detection. In *CVPR*, 2017.

[72] J. Zhang, K. J. Shih, A. Elgammal, A. Tao, and B. Catanzaro. Graphical contrastive losses for scene graph parsing. In *CVPR*, 2019.