Words Aren't Enough, Their Order Matters: On the Robustness of Grounding Visual Referring Expressions

Arjun R. Akula[1]*, **Spandana Gella**[2], **Yaser Al-Onaizan**[2], **Song-Chun Zhu**[1], **Siva Reddy**[3]
[1]UCLA Center for Vision, Cognition, Learning, and Autonomy, [2]Amazon AI
[3]Facebook CIFAR AI Chair, Mila; McGill University
aakula@ucla.edu, sgella@amazon.com, onaizan@amazon.com,
sczhu@stat.ucla.edu, siva.reddy@mila.quebec

Abstract

Visual referring expression recognition is a challenging task that requires natural language understanding in the context of an image. We critically examine *RefCOCOg*, a standard benchmark for this task, using a human study and show that 83.7% of test instances do not require reasoning on linguistic structure, i.e., words are enough to identify the target object, the word order doesn't matter. To measure the true progress of existing models, we split the test set into two sets, one which requires reasoning on linguistic structure and the other which doesn't. Additionally, we create an out-of-distribution dataset *Ref-Adv* by asking crowdworkers to perturb in-domain examples such that the target object changes. Using these datasets, we empirically show that existing methods fail to exploit linguistic structure and are 12% to 23% lower in performance than the established progress for this task. We also propose two methods, one based on contrastive learning and the other based on multi-task learning, to increase the robustness of ViLBERT, the current state-of-the-art model for this task. Our datasets are publicly available at https://github.com/aws/aws-refcocog-adv.

1 Introduction

Visual referring expression recognition is the task of identifying the object in an image referred by a natural language expression (Kazemzadeh et al., 2014; Nagaraja et al., 2016; Mao et al., 2016; Hu et al., 2016). Figure 1 shows an example. This task has drawn much attention due to its ability to test a model's understanding of natural language in the context of visual grounding and its application in downstream tasks such as image retrieval (Young et al., 2014) and question answering (Antol et al., 2015; Zhu et al., 2016). To track progress on this task, various datasets have been proposed, in which real world images are annotated by crowdsourced workers (Kazemzadeh et al., 2014; Mao et al., 2016). Recently, neural models have achieved tremendous progress on these datasets (Yu et al., 2018; Lu et al., 2019). However, multiple studies have suggested that these models could be exploiting strong biases in these datasets (Cirik et al., 2018b; Liu et al., 2019). For example, models could be just selecting a salient object in an image or a referring expression without recourse to linguistic structure (see Figure 1). This defeats the true purpose of the task casting doubts on the actual progress.

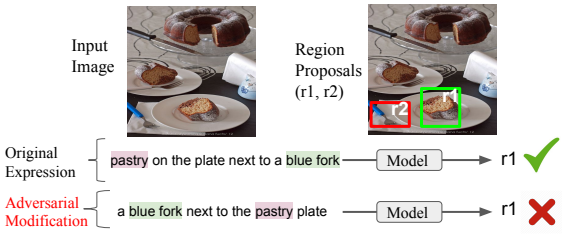

Figure 1: An example of the visual referring expression recognition task. If the word *pastry* is present in the referring expression, models prefer the bounding box *r1* (highlighted in green) irrespective of the change in linguistic structure (word order).

In this work, we examine *RefCOCOg* dataset (Mao et al., 2016), a popular testbed for evaluating referring expression models, using crowdsourced workers. We show that a large percentage of samples in the *RefCOCOg* test set indeed do not rely on linguistic structure (word order) of the expressions. Accordingly, we split *RefCOCOg* test set into two splits, *Ref-Easy* and *Ref-Hard*, where linguistic structure is key for recognition in the latter but not the former (§2). In addition, we create a new out-of-distribution[1] dataset called *Ref-Adv* using *Ref-Hard* by rewriting a referring expression

*Work done in part while AA was intern at Amazon AI.

[1]This is a *contrast set* according to Gardner et al. (2020)

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 6555–6565
July 5 - 10, 2020.

such that the target object is different from the original annotation (§3). We evaluate existing models on these splits and show that the true progress is at least 12-23% behind the established progress, indicating there is ample room for improvement (§4). We propose two new models, one which make use of contrastive learning using negative examples, and the other based on multi-task learning, and show that these are slightly more robust than the current state-of-the-art models (§5).

2 Importance of linguistic structure

RefCOCOg is the largest visual referring expression benchmark available for real world images (Mao et al., 2016). Unlike other referring expression datasets such as *RefCOCO* and *RefCOCO+* (Kazemzadeh et al., 2014), a special care has been taken such that expressions are longer and diverse. We therefore choose to examine the importance of linguistic structure in *RefCOCOg*.

Cirik et al. (2018b) observed that when the words in a referring expression are shuffled in random order, the performance of existing models on *RefCOCOg* drops only a little. This suggests that models are relying heavily on the biases in the data than on linguistic structure, i.e., the actual sequence of words. Ideally, we want to test models on samples where there is correlation between linguistic structure and spatial relations of objects, and any obscurity in the structure should lead to ambiguity. To filter out such set, we use humans.

We randomly shuffle words in a referring expression to distort its linguistic structure, and ask humans to identify the target object of interest via predefined bounding boxes. Each image in *RefCOCOg* test set is annotated by five Amazon Mechanical Turk (AMT) workers and when at least three annotators select a bounding box that has high overlap with the ground truth, we treat it as a correct prediction. Following Mao et al. (2016), we set 0.5 IoU (intersection over union) as the threshold for high overlap. Given that there are at least two objects in each image, the optimal performance of a random choice is less than 50%.[2] However, we observe that human accuracy on distorted examples is 83.7%, indicating that a large portion of *RefCOCOg* test set is insensitive to linguistic structure. Based on this observation, we divide the test set into two splits for fine-grained evaluation of models: ***Ref-Easy*** contains samples insensitive to linguistic structure and ***Ref-Hard*** contains sensitive samples (statistics of the splits are shown in Table 1).

[2]On average, there are 8.2 bounding boxes per image.

	Ref-Easy	*Ref-Hard*	*Ref-Adv*
data size	8034 (83.7% of *RefCOCOg*)	1568 (16.3% of *RefCOCOg*)	3704
avg. length in words	8.0	10.2	11.4

Table 1: Statistics of *Ref-Easy*, *Ref-Hard* and *Ref-Adv*. *Ref-Easy* and *Ref-Hard* indicate the proportion of samples in *RefCOCOg* test set that are insensitive and sensitive to linguistic structure respectively.

3 An out-of-distribution dataset

Due to unintended annotation artifacts in *RefCOCOg*, it is still possible that models could perform well on *Ref-Hard* without having to rely on linguistic structure, e.g., by selecting frequent objects seen during training time. Essentially, *Ref-Hard* is an in-distribution split. To avoid this, we create ***Ref-Adv***, an adversarial test set with samples that may be fall out of training distribution.

We take each sample in *Ref-Hard* and collect additional referring expressions such that the target object is different from the original object. We chose the target objects which humans are most confused with when the referring expression is shuffled (as described in the previous section). For each target object, we ask three AMT workers to write a referring expression while retaining most content words in the original referring expression. In contrast to the original expression, the modified expression mainly differs in terms of the structure while sharing several words. For example, in Figure 1, the adversarial sample is created by swapping *pastry* and *blue fork* and making *plate* as the head of *pastry*. We perform an extra validation step to filter out bad referring expressions. In this step, three additional AMT workers select a bounding box to identify the target object, and we only select the samples where at least two workers achieve IoU > 0.5 with the target object.

Since the samples in *Ref-Adv* mainly differ in linguistic structure with respect to *Ref-Hard*, we hope that a model which does not make use of linguistic structure (and correspondingly spatial relations between objects) performs worse on *Ref-Adv* even when it performs well on *Ref-Hard* due to exploiting biases in the training data.

Figure 2 shows several examples from the *Ref-Easy*, *Ref-Hard*, and *Ref-Adv* splits. We note that

Easy: A dinning table with cake and drinks
Hard: A chair with a purse hanging from it
Adv: The purse which is hanging from a chair

Easy: Bus
Hard: Bus in the middle of the crowd
Adv: The crowd that the bus is in the middle of

Easy: The larger of two giraffes
Hard: A giraffe eating leaves off the tree
Adv: The giraffe that is not eating leaves off the tree

Easy: A blue snowboard
Hard: A woman wearing a blue jacket and orange glasses next to a woman with a white hood
Adv: A woman with a white hood, next to a woman wearing orange glasses and a blue jacket.

Easy: Water in a tall, clear glass
Hard: The glass of water next to the saucer with the cup on it
Adv: The cup on the saucer, next to the glass of water

Easy: The short blue bike on the right
Hard: The blue bike behind the red car
Adv: The red car behind the blue bike

Easy: The man with the glasses on
Hard: A man holding a cake that is not wearing a tie
Adv: The man holding a cake that is wearing a tie

Easy: A green cushion couch with a pillow
Hard: A green couch across from a white couch
Adv: A white couch across from a green couch

Figure 2: Examples from *Ref-Easy*, *Ref-Hard*, and *Ref-Adv* splits. As seen, *Ref-Hard* and *Ref-Adv* have several words in common but differ in their linguistic structure and the target object of interest.

Ref-Adv expressions are longer on average than *Ref-Easy* and *Ref-Hard* (Figure 6 in appendix) and consists of rich and diverse spatial relationships (Figure 7 in appendix).

Concurrent to our work, Gardner et al. (2020) also propose perturbed test splits for several tasks by modifying in-domain examples. In their setup, the original authors of each task create perturbed examples, whereas we use crowdworkers. Closest to our work is from Kaushik et al. (2020) who also use crowdworkers. While we use perturbed examples to evaluate robustness, they also use them to improve robustness (we propose complementary methods to improve robustness §5). Moreover, we are primarily concerned with the robustness of models for visual expression recognition task, while Gardner et al. and Kaushik et al. focus on different tasks (e.g., sentiment, natural language inference).

3.1 Human Performance on *Ref-Easy*, *Ref-Hard* and *Ref-Adv*

We conducted an additional human study (on AMT) to compare the human performance on *Ref-Easy*, *Ref-Hard* and *Ref-Adv* splits. First, we randomly sampled 100 referring expressions from each of the three splits. Each referring expression is then assigned to three AMT workers and are asked to select a bounding box to identify the target object. We considered a sample to be correctly annotated by humans if at least two out of three workers select the ground-truth annotation. Through this evaluation, we obtained human performance on each of the three splits Ref-Easy, Ref-Hard, and Ref-Adv as 98%, 95%, and 96% respectively.

4 Diagnosing Referring Expression Recognition models

We evaluate the following models, most of which are designed to exploit linguistic structure.

CMN (Compositional Modular Networks; Hu et al. 2017; Andreas et al. 2016) grounds expressions using neural modules by decomposing an expression into <subject, relation, object> triples. The subject and object are localized to the objects in the image using a localization module while the relation between them is modeled using a relationship module. The full network learns to jointly decompose the input expression into a triple while also recognizing the target object.

GroundNet (Cirik et al., 2018a) is similar to CMN, however it makes use of rich linguistic structure (and correspondingly rich modules) as defined by an external syntactic parser.

MattNet (Yu et al., 2018) generalizes CMN to flexibly adapt to expressions that cannot be captured by the fixed template of CMN. It introduces new modules and also uses an attention mechanism to weigh modules.

ViLBERT (Lu et al., 2019), the state-of-the-art model for referring expression recognition, uses a

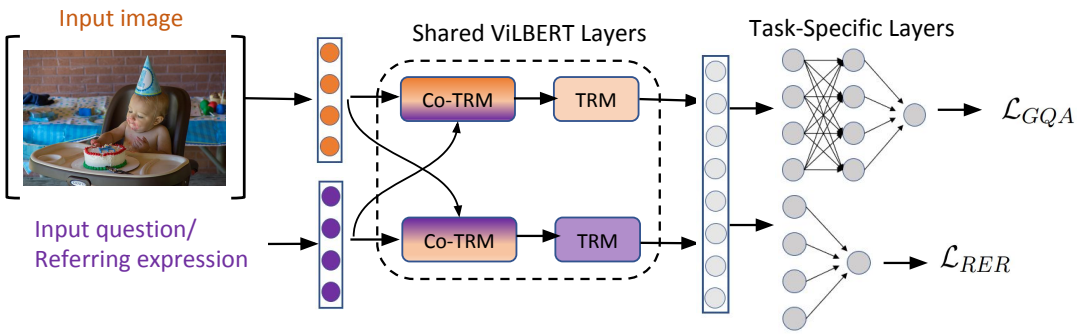

Figure 3: Multi-task learning model for referring expression recognition with GQA

pretrain-then-transfer learning approach to jointly learn visiolinguistic representations from large-scale data and utilizes them to ground expressions. This is the only model that does not explicitly model compositional structure of language, but BERT-like models are shown to capture syntactic structure latently (Hewitt and Manning, 2019).

4.1 Results and discussion

We trained on the full training set of *RefCOCOg* and performed hyperparameter tuning on a development set. We used the development and test splits of Mao et al. (2016). Table 2 shows the model accuracies on these splits and our proposed datasets. The models are trained to select ground truth bounding box from a set of predefined bounding boxes. We treat a prediction as positive if the predicted bounding box has IoU > 0.5 with the ground truth.

Although the overall performance on the test set seem high, in reality, models excel only at *Ref-Easy* while performing poorly on *Ref-Hard*. The difference in performance between *Ref-Easy* and *Ref-Hard* ranges up to 15%. This indicates that current models do not exploit linguistic structure effectively. When tested on *Ref-Adv*, the performance goes down even further, increasing the gap between *Ref-Easy* and *Ref-Adv* (up to 26%). This suggests that models are relying on reasoning shortcuts found in training than actual understanding. Among the models, GroundNet performs worse, perhaps due to its reliance on rigid structure predicted by an external parser and the mismatches between the predicted structure and spatial relations between objects. ViLBERT achieves the highest performance and is relatively more robust than other models. In the next section, we propose methods to further increase the robustness of ViLBERT.

Model	*Dev*	*Test*	*Easy*	*Hard*	*Adv*
GroundNet	66.50	65.80	67.11	54.47	42.90
CMN	70.00	69.40	69.55	68.63	49.50
MattNet	79.21	78.51	80.96	65.94	54.64
ViLBERT	**83.39**	**83.63**	**85.93**	**72.00**	**70.90**

Table 2: Accuracy of models on *RefCOCOg* standard splits and our splits *Ref-Easy*, *Ref-Hard* and *Ref-Adv*.

5 Increasing the robustness of ViLBERT

We extend ViLBERT in two ways, one based on contrastive learning using negative samples, and the other based on multi-task learning on GQA (Hudson and Manning, 2019), a task that requires linguistic and spatial reasoning on images.

Contrastive learning using negative samples Instead of learning from one single example, contrastive learning aims to learn from multiple examples by comparing one to the other. In order to increase the sensitivity to linguistic structure, we mine negative examples that are close to the current example and learn to jointly minimize the loss on the current (positive) example and maximize the loss on negative examples. We treat the triplets (i, e, b) in the training set as positive examples, where i, e, b stands for image, expression and ground truth bounding box. For each triplet (i, e, b), we sample another training example (i', e', b'), and use it to create two negative samples, defined by (i', e, b') and (i, e', b), i.e., we pair wrong bounding boxes with wrong expressions. For efficiency, we only consider negative pairs from the mini-batch. We modify the batch loss function as follows:

$$\mathcal{L}(\mathbf{i}, \mathbf{e}, \mathbf{b}) = \mathbf{F}_{(\mathbf{e},\mathbf{e}')} \left[\ell(\mathbf{i}, \mathbf{e}, \mathbf{b}) - \ell(\mathbf{i}, \mathbf{e}', \mathbf{b}) - \tau\right]_+ + \mathbf{F}_{(\mathbf{i},\mathbf{i}')} \left[\ell(\mathbf{i}, \mathbf{e}, \mathbf{b}) - \ell(\mathbf{i}', \mathbf{e}, \mathbf{b}') - \tau\right]_+$$

Model	*Dev*	*Test*	*Easy*	*Hard*	*Adv*
ViLBERT (VB)	83.39	83.63	85.93	72.00	70.90
VB+*Sum-H*	81.61	83.00	85.93	70.60	72.30
VB+*Max-H*	82.93	82.70	**86.58**	70.46	73.35
VB+*MTL (GQA)*	**83.45**	**84.30**	86.23	**73.79**	**73.92**

Table 3: Accuracy of enhanced ViLBERT models.

Here $\ell(i, e, b)$ is the cross-entropy loss of ViLBERT, $[x]_+$ is the hinge loss defined by $\max(0, x)$, and τ is the margin parameter. F indicates a function over all batch samples. We define F to be either sum of hinges (Sum-H) or max of hinges (Max-H). While Sum-H takes sum over all negative samples, If batch size is n, for each (i, e, b), there will be $n-1$ triplets of (i', e, b') and (i, e', b). For (i, e, b), there will be one (i', e, b') and one (i, e', b). Similar proposals are known to increase the robustness of vision and language problems like visual-semantic embeddings and image description ranking (Kiros et al., 2014; Gella et al., 2017; Faghri et al., 2018).

Multi-task Learning (MTL) with GQA In order to increase the sensitivity to linguistic structure, we rely on tasks that require reasoning on linguistic structure and learn to perform them alongside our task. We employ MTL with GQA (Hudson and Manning, 2019), a compositional visual question answering dataset. Specifically, we use the GQA-Rel split which contains questions that require reasoning on both linguistic structure and spatial relations (e.g., *Is there a boy wearing a red hat standing next to yellow bus?* as opposed to *Is there a boy wearing hat?*). Figure 3 depicts the neural architecture. We share several layers between the tasks to enable the model to learn representations useful for both tasks. Each shared layer constitute a co-attention transformer block (Co-TRM; Lu et al. 2019) and a transformer block (TRM; Vaswani et al. 2017). While in a transformer, attention is computed using queries and keys from the same modality, in a co-attention transformer they come from different modalities (see cross arrows in Figure 3). The shared representations are eventually passed as input to task-specific MLPs. We optimize each task using alternative training (Luong et al., 2015).

Results and discussion Table 3 shows the experimental results on the referring expression recognition task. Although contrastive learning improves the robustness of ViLBERT on *Ref-Adv* (+1.4% and +2.5% for Sum-H and Max-H respectively), it comes at a cost of slight performance drop on the full test (likely due to sacrificing biases shared between training and test sets). Whereas MTL improves the robustness on all sets showing that multi-task learning helps (we observe 2.3% increase on GQA §A.5.2). Moreover, the performance of MTL on *Ref-Hard* and *Ref-Adv* are similar, suggesting that the model generalizes to unseen data distribution. Figure 4 shows qualitative examples comparing MTL predictions on *Ref-Hard* and *Ref-Adv* parallel examples. These suggest that the MTL model is sensitive to linguistic structure. However, there is still ample room for improvement indicated by the gap between *Ref-Easy* and *Ref-Hard* (12.4%).

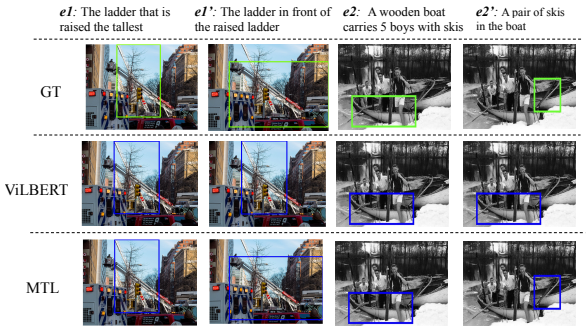

Figure 4: Predictions of ViLBERT and MTL model (GT denotes ground-truth). $e1'$ and $e2'$ are adversarial expressions of $e1$ and $e2$ respectively.

6 Conclusion

Our work shows that current datasets and models for visual referring expressions fail to make effective use of linguistic structure. Although our proposed models are slightly more robust than existing models, there is still significant scope for improvement. We hope that *Ref-Hard* and *Ref-Adv* will foster more research in this area.

Acknowledgements

We would like to thank Volkan Cirik, Licheng Yu, Jiasen Lu for their help with GroundNet, MattNet and ViLBERT respectively, Keze Wang for his help with technical issues, and AWS AI data team for their help with Mechanical Turk. We are grateful to the anonymous reviewers for their useful feedback.

References

Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Dan Klein. 2016. Neural module networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 39–48.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. 2015. Vqa: Visual question answering. In *IEEE international conference on computer vision*, pages 2425–2433.

Volkan Cirik, Taylor Berg-Kirkpatrick, and Louis-Philippe Morency. 2018a. Using syntax to ground referring expressions in natural images. In *AAAI Conference on Artificial Intelligence*.

Volkan Cirik, Louis-Philippe Morency, and Taylor Berg-Kirkpatrick. 2018b. Visual referring expression recognition: What do systems actually learn? In *North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT*, pages 781–787.

Fartash Faghri, David J. Fleet, Jamie Ryan Kiros, and Sanja Fidler. 2018. VSE++: improving visual-semantic embeddings with hard negatives. In *British Machine Vision Conference*, page 12.

Matt Gardner, Yoav Artzi, Victoria Basmova, Jonathan Berant, Ben Bogin, Sihao Chen, Pradeep Dasigi, Dheeru Dua, Yanai Elazar, Ananth Gottumukkala, Nitish Gupta, Hanna Hajishirzi, Gabriel Ilharco, Daniel Khashabi, Kevin Lin, Jiangming Liu, Nelson F. Liu, Phoebe Mulcaire, Qiang Ning, Sameer Singh, Noah A. Smith, Sanjay Subramanian, Reut Tsarfaty, Eric Wallace, Ally Zhang, and Ben Zhou. 2020. Evaluating nlp models via contrast sets. *arXiv preprint arXiv:2004.02709*.

Spandana Gella, Rico Sennrich, Frank Keller, and Mirella Lapata. 2017. Image pivoting for learning multilingual multimodal representations. In *Empirical Methods in Natural Language Processing*, pages 2839–2845.

John Hewitt and Christopher D Manning. 2019. A structural probe for finding syntax in word representations. In *North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4129–4138.

Ronghang Hu, Marcus Rohrbach, Jacob Andreas, Trevor Darrell, and Kate Saenko. 2017. Modeling relationships in referential expressions with compositional modular networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1115–1124.

Ronghang Hu, Huazhe Xu, Marcus Rohrbach, Jiashi Feng, Kate Saenko, and Trevor Darrell. 2016. Natural language object retrieval. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4555–4564.

Drew A Hudson and Christopher D Manning. 2019. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 6700–6709.

Divyansh Kaushik, Eduard Hovy, and Zachary Lipton. 2020. Learning the difference that makes a difference with counterfactually-augmented data. In *International Conference on Learning Representations*.

Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. 2014. Referitgame: Referring to objects in photographs of natural scenes. In *Empirical methods in natural language processing (EMNLP)*, pages 787–798.

Ryan Kiros, Ruslan Salakhutdinov, and Richard S Zemel. 2014. Unifying visual-semantic embeddings with multimodal neural language models. *arXiv preprint arXiv:1411.2539*.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. 2017. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123(1):32–73.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer.

Runtao Liu, Chenxi Liu, Yutong Bai, and Alan L. Yuille. 2019. Clevr-ref+: Diagnosing visual reasoning with referring expressions. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019*, pages 4185–4194.

Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *arXiv preprint arXiv:1908.02265*.

Minh-Thang Luong, Quoc V Le, Ilya Sutskever, Oriol Vinyals, and Lukasz Kaiser. 2015. Multi-task sequence to sequence learning. *arXiv preprint arXiv:1511.06114*.

Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L Yuille, and Kevin Murphy. 2016. Generation and comprehension of unambiguous object descriptions. In *IEEE conference on computer vision and pattern recognition*, pages 11–20.

Varun K. Nagaraja, Vlad I. Morariu, and Larry S. Davis. 2016. Modeling context between objects for referring expression understanding. In *European Conference on Computer Vision*, pages 792–807.

Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. 2018. Conceptual captions: A

cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Association for Computational Linguistics*, pages 2556–2565.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. 2014. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78.

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

Yuke Zhu, Oliver Groth, Michael Bernstein, and Li Fei-Fei. 2016. Visual7w: Grounded question answering in images. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4995–5004.

A Appendix

In this supplementary material, we begin by providing more details on *RefCOCOg* dataset to supplement Section 2 of the main paper. We then provide *Ref-Adv* annotation details, statistics, analysis, and random examples, to supplement Section 3 of the main paper. Finally, we provide details of our models (initialization & training, hyper-parameters) and show additional results to supplement Section 5 of the main paper.

A.1 *RefCOCOg* vs Other Referring Expressions Datasets

RefCOCO, *RefCOCO+* (Kazemzadeh et al., 2014) and *RefCOCOg* (Google-RefCOCO; Mao et al. 2016) are three commonly studied visual referring expression recognition datasets for real images. All the three data sets are built on top of MSCOCO dataset (Lin et al., 2014) which contains more than 300,000 images, with 80 categories of objects. *RefCOCO*, *RefCOCO+* were collected using online interactive game. *RefCOCO* dataset is more biased towards person category. *RefCOCO+* does not allow the use of location words in the expressions, and therefore contains very few spatial relationships. *RefCOCOg* was not collected in an interactive setting and therefore contains longer expressions.

For our adversarial analysis, we chose *RefCOCOg* for the following three important reasons: Firstly, expressions are longer (by 2.5 times on average) in *RefCOCOg* and therefore contains more spatial relationships compared to other two datasets. Secondly, *RefCOCOg* contains at least 2 to 4 instances of the same object type within the same image referred by an expression. This makes the dataset more robust, and indirectly puts higher importance on grounding spatial relationships in finding the target object. Finally, as shown in Table 4, *RefCOCO* and *RefCOCO+* are highly skewed towards *Person* object category ($\approx 50\%$) whereas *RefCOCOg* is relatively less skewed ($\approx 36\%$), more diverse, and less biased.

A.2 Importance of Linguistic Structure

Cirik et al. (2018b) observed that existing models for *RefCOCOg* are relying heavily on the biases in the data than on linguistic structure. We perform extensive experiments to get more detailed insights into this observation. Specifically, we distort linguistic structure of referring expressions in the *Re-*

	RefCOCO	*RefCOCO+*	*RefCOCOg*
Outdoor	0.89%	0.88%	1.65%
Food	10.16%	10.07%	8.10%
Indoor	3.10%	3.09%	2.59%
Appliance	0.67%	0.68%	1.03%
Kitchen	3.95%	3.95%	5.40%
Accessory	2.33%	2.33%	2.85%
Person	49.50%	49.70%	37.02%
Animal	13.26%	13.27%	15.05%
Vehicle	7.23%	7.22%	10.71%
Sports	0.73%	0.74%	1.91%
Electronic	1.94%	1.95%	2.56%
Furniture	6.14%	6.12%	11.09%

Table 4: Distribution of object categories in *RefCOCO*, *RefCOCO+*, and *RefCOCOg* datasets.

fCOCOg test split and evaluate the SOTA models that are trained on original undistorted *RefCOCOg* training split. Similar to (Cirik et al., 2018b), we distort the test split using two methods: (a) randomly shuffle words in a referring expression, and (b) delete all the words in the expression except for nouns and adjectives. Table 5 shows accuracies for the models with (column 3 and 4) and without (column 2) distorted referring expressions. Except for the ViLBERT model(Lu et al., 2019), the drop in accuracy is not significant indicating that spatial relations are ignored in grounding the referring expression.

Using the relatively robust ViLBERT model, we repeat this analysis on our splits *Ref-Easy*, *Ref-Hard* and *Ref-Adv*. We randomly sampled 1500 expressions from each of these splits and then compare performance of ViLBERT on these three sets. As shown in Table 6, we find a large difference in model's accuracy on *Ref-Hard* and *Ref-Adv*. This clearly indicates that grounding expressions in both of these splits require linguistic and spatial reasoning.

A.3 *Ref-Adv* Annotation

We construct *Ref-Adv* by using all the 9602 referring expressions from *RefCOCOg* test data split. As shown in Figure 5, we follow a three stage approach to collect these new samples:

Stage 1: For every referring expression in *RefCOCOg* test split, we perturb its linguistic structure by shuffling the word order randomly. We show each of these perturbed expression along with images and all object bounding boxes to five qualified Amazon Mechanical Turk (AMT) workers and ask them to identify the ground-truth bounding box for the shuffled referring expression. We hired workers from US and Canada with approval rates higher than 98% and more than 1000 accepted HITs. At the beginning of the annotation, we ask the turkers to go through a familiarization phase where they become familiar with the task. We consider all the image and expression pairs for which at least 3 out of 5 annotators **failed to locate** the object correctly (with IoU < 0.5) as hard samples (***Ref-Hard***). We refer to the image-expressions for which at least 3 out of 5 annotators were **able to localize** the object correctly as easy samples (***Ref-Easy***). On average, we found that humans failed to localize the objects correctly in 17% of the expressions.

Model	Original	Shuf	N+J
CMN (Hu et al., 2017)	69.4	66.4	67.4
GroundNet (Cirik et al., 2018a)	65.8	57.6	62.8
MattNet (Yu et al., 2018)	78.5	75.3	76.1
ViLBERT (Lu et al., 2019)	83.6	71.4	73.6

Table 5: *RefCOCOg* test accuracies of SOTA models on (a) original undistorted split, (b) after randomly shuffling words (Shuf) in the referring expression, and (c) after deleting all the words except for nouns and adjectives (N+J). ViLBERT is relatively more robust than other baselines.

Test	Original	Shuf	N+J
Ref-Easy	86.40	75.06	76.00
Ref-Hard	72.73	51.13	56.60
Ref-Adv	71.08	50.23	57.40

Table 6: *Ref-Easy*, *Ref-Hard*, and *Ref-Adv* test accuracies of ViLBERT on (a) original undistorted split, (b) after randomly shuffling words (Shuf) in the referring expression, and (c) after deleting all the words except for nouns and adjectives (N+J).

Stage 2: We take *Ref-Hard* images and ask turkers to generate adversarial expressions such that the target object is different from the original object. More concretely, for each of the hard samples, we identify the most confused image regions among human annotators as the target objects in stage 1. For each of these target objects, we then ask three

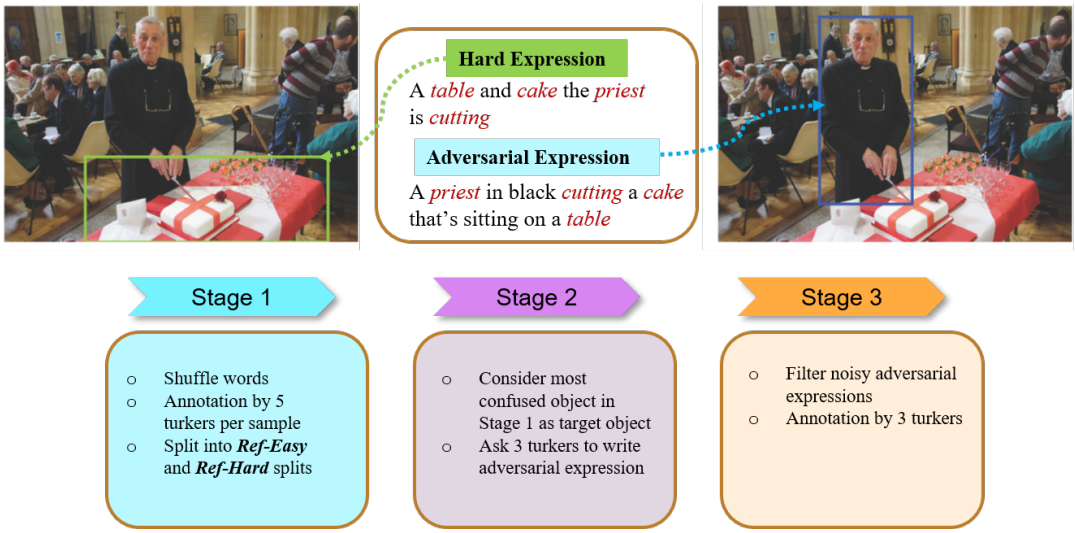

Figure 5: Overview of our three-stage *Ref-Adv* construction process. Given the image, referring expression, ground-truth bounding boxes for all the samples in *RefCOCOg* test split, we first filter out the hard samples and then construct adversarial expressions using them. Please refer to section 2 for further detail.

Referring Expressions	3704
Unique Images	976
Vocabulary	2319
Avg. Length of Expression	11.4

Table 7: *Ref-Adv* Statistics

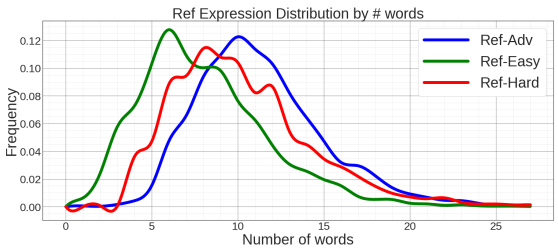

Figure 6: Referring expression length distribution for *Ref-Easy*, *Ref-Hard*, *Ref-Adv* datasets.

turkers to write a referring expression while retaining at least three content words (nouns and adjectives) in the original referring expression. This generates adversarial expressions for the original ground-truth *Ref-Hard* referring expressions.

Stage 3: We filter out the noisy adversarial expressions generated in stage 2 by following a validation routine used in the generation of *RefCOCOg* dataset. We ask three additional AMT workers to select a bounding box to identify the target object in the adversarial expression and then remove the noisy samples for which the inter-annotator agreement among workers is low. The samples with at least 2 out of 3 annotators achieving IoU > 0.5 will be added to *Ref-Adv* dataset.

A.4 Dataset Analysis, Comparison, and Visualization

In Table 7 we summarize the size and complexity of our *Ref-Adv* split. Figure 6 shows expression length distribution of *Ref-Easy*, *Ref-Hard*, and *Ref-Adv*. It should be noted that *Ref-Adv* expressions are longer on average than *Ref-Easy* and *Ref-Hard*.

Distribution of object categories in *Ref-Easy*, *Ref-Hard* and *Ref-Adv* is shown in Table 8. In comparison to *Ref-Easy* and *Ref-Hard*, *Ref-Adv* is more balanced and less biased towards `Person` category. Figure 7 shows the relative frequency of the most frequent spatial relationships in all the three splits. As we can see, *Ref-Adv* comprises of rich and diverse spatial relationships. In Table 2, we show random selection of the *Ref-Easy*, *Ref-Hard*, and *Ref-Adv* splits.

A.5 Model and other Experiment Details

A.5.1 Datasets

GQA (Hudson and Manning, 2019) contains 22M questions generated from Visual Genome (Krishna et al., 2017) scene graphs. However, in our our multi-task training (MTL), we leverage only 1.42M questions that require reasoning on both linguistic structure and spatial relations. We filter these re-

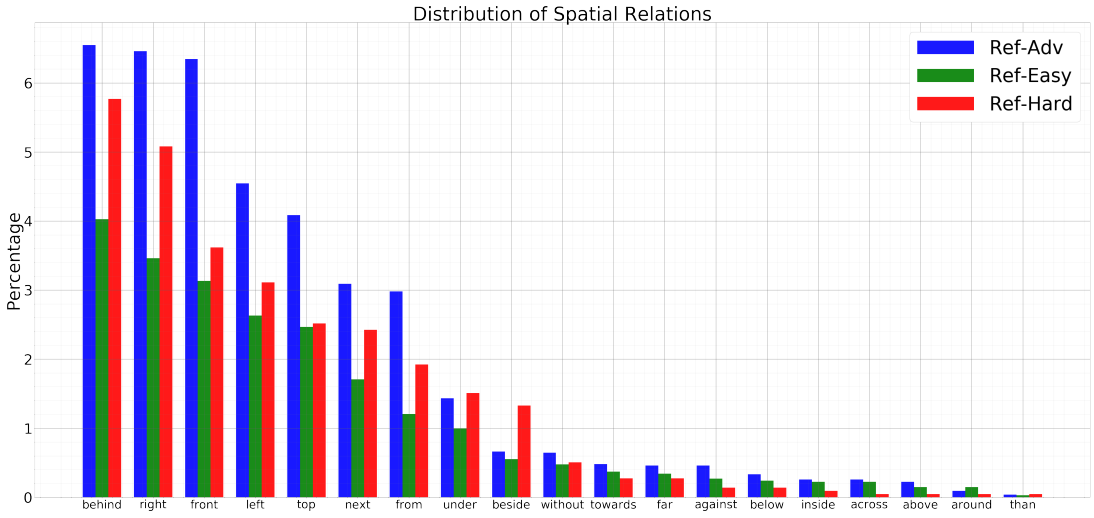

Figure 7: Relative frequency of the most frequent spatial relationships in *Ref-Easy*, *Ref-Hard*, and *Ref-Adv*

	Ref-Easy 8034 samples	*Ref-Hard* 1568 samples	*Ref-Adv* 3704 samples
Outdoor	1.21%	1.90%	1.97%
Food	7.94%	9.80%	9.63%
Indoor	2.81%	2.83%	2.76%
Appliance	0.80%	1.07%	1.11%
Kitchen	4.52%	5.73%	5.77%
Accessory	3.20%	5.44%	5.29%
Person	37.26%	20.88%	21.01%
Animal	15.95%	13.92%	13.90%
Vehicle	10.91%	10.40%	10.26%
Sports	1.45%	5.04%	5.13%
Electronic	2.62%	3.20%	3.31%
Furniture	11.28%	19.73%	19.83%

Table 8: Distribution of object categories in *Ref-Easy*, *Ref-Hard*, and *Ref-Adv* splits.

lational questions by applying the following constraint on question types: *type.Semantic=*'***rel***'. We also apply this constraint for filtering the development set. We denote this subset as *GQA-Rel*. We considered GQA-Rel instead of GQA for two reasons: 1) GQA-Rel is a more related task to RefCOCOg; and 2) MTL training with the full GQA set is computationally expensive. For each question in the dataset, there exists a long answer (free-form text) and a short answer (containing one or two words). We only consider the short answers for the questions and treat the unique set of answers as output categories. While the full GQA dataset has 3129 output categories, GQA-Rel contains only 1842 categories.

We follow Yu et al. (2018) in creating the train (80512 expressions), val (4896 expressions), and test (9602 expressions) splits of ***RefCOCOg***. For all our experiments in this paper, we directly use the ground-truth bounding box proposals.

A.5.2 Training

ViLBERT Pre-training We used pre-trained ViLBERT model that is trained on 3.3 million image-caption pairs from Conceptual Captions dataset (Sharma et al., 2018).[3]

Single-Task Fine-tuning on *RefCOCOg* In order to fine-tune the baseline ViLBERT (Lu et al., 2019) model on *RefCOCOg* dataset, we pass the ViLBERT visual representation for each bounding box into a linear layer to predict a matching score (similar to RefCOCO+ training in Lu et al. 2019). We calculate accuracy using IoU metric (prediction is correct if IoU(predicted_region, ground-truth region) > 0.5). We use a binary cross-entropy loss and train the model for a maximum of 25 epochs. We use early-stopping based on the validation performance. We use an initial learning rate of 4e-5 and use a linear decay learning rate schedule with warm up. We train on 8 Tesla V100 GPUs with a total batch size of 512.

Negative Mining We used a batch size of 512 and randomly sample negatives from the mini-batch for computational efficiency. We sampled 64 negatives from each batch for both Sum of Hinges and Max of Hinges losses. We fine-tune the margin

[3]ViLBERT 8-Layer model at the link https://github.com/jiasenlu/vilbert_beta

Split	Before MTL	After MTL
GQA-Rel Dev	53.7%	56.0%
GQA Dev	40.24%	42.1%
GQA Test	36.64%	39.2%

Table 9: Performance on GQA-Rel Dev, GQA-Dev and GQA-Test splits *before* and *after* MTL training with *RefCOCOg* (Note: MTL training for all the three rows is performed using GQA-Rel and *RefCOCOg*).

ViLBERT	*Ref-Dev*	*Ref-Test*	*Ref-Adv*
Without TL and MTL	83.39	83.63	70.90
TL with VQA	82.26	84.14	72.96
TL with GQA	80.60	82.08	70.41
TL with GQA-Rel	81.05	83.12	70.78
MTL with VQA	81.20	82.10	70.82
MTL with GQA-Rel	**83.45**	**84.30**	**73.92**

Table 10: Comparing ViLBERT's Multi-task Learning (MTL) with Transfer Learning (TL) experiments. *Ref-Dev* and *Ref-Test* correspond to: *RefCOCOg-Dev* and *RefCOCOg-Test* splits respectively.

parameters based on development split. We train the model for a maximum of 25 epochs. We use early-stopping based on the validation performance. We use an initial learning rate of 4e-5 and use a linear decay learning rate schedule with warm up. We train on 8 Tesla V100 GPUs with a total batch size of 512.

Multi-Task Learning (MTL) with GQA-Rel The multi-task learning architecture is shown in Figure 3 in the main paper. The shared layers constitute transformer blocks (TRM) and co-attentional transformer layers (Co-TRM) in ViLBERT (Lu et al., 2019). The task-specific layer for GQA task is a two-layer MLP and we treat it as a multi-class classification task and the task-specific layer for RER is a linear layer that predicts a matching score for each of the image regions given an input referring expression. The weights for the task-specific layers are randomly initialized, whereas the shared layers are initialized with weights pre-trained on 3.3 million image-caption pairs from Conceptual Captions dataset (Sharma et al., 2018). We use a binary cross-entropy loss for both tasks. Similar to Luong et al. (2015), during training, we optimize each task alternatively in mini-batches based on a mixing ratio. We use early-stopping based on the validation performance. We use an initial learning rate of 4e-5 for *RefCOCOg* and 2e-5 for GQA, and use a linear decay learning rate schedule with warm up. We train on 4 RTX 2080 GPUs with a total batch size of 256.

GQA MTL Results Table 3 in the main paper showed that MTL training with GQA-Rel significantly improved the performance of model on *Ref-Hard* and *Ref-Adv* splits. In addition, we also observed a significant improvement in GQA-Rel development, GQA development and test splits as shown in the Table 9.

A.5.3 Additional Experiments

In this subsection, we present results of additional experiments using transfer learning (TL) and multi-task learning (MTL) with ViLBERT on VQA, GQA, and GQA-Rel tasks. As shown in Table 10, TL with VQA showed slight improvement. However, TL with GQA, TL with GQA-Rel, and MTL with VQA did not show any improvements [4].

[4] We could not perform MTL with GQA as it requires large number of computational resources.