Flexible Visual Grounding

Yongmin Kim **Chenhui Chu** **Sadao Kurohashi**
Kyoto University, Kyoto, Japan
{yongmin, chu, kuro}@nlp.ist.i.kyoto-u.ac.jp

Abstract

Existing visual grounding datasets are artificially made, where every query regarding an entity must be able to be grounded to a corresponding image region, i.e., answerable. However, in real-world multimedia data such as news articles and social media, many entities in the text cannot be grounded to the image, i.e., unanswerable, due to the fact that the text is unnecessarily directly describing the accompanying image. A robust visual grounding model should be able to flexibly deal with both answerable and unanswerable visual grounding. To study this flexible visual grounding problem, we construct a pseudo dataset and a social media dataset including both answerable and unanswerable queries. In order to handle unanswerable visual grounding, we propose a novel method by adding a pseudo image region corresponding to a query that cannot be grounded. The model is then trained to ground to ground-truth regions for answerable queries and pseudo regions for unanswerable queries. In our experiments, we show that our model can flexibly process both answerable and unanswerable queries with high accuracy on our datasets.[1]

1 Introduction

Starting from conventional vision-and-language tasks such as image captioning (Vinyals et al., 2015) and visual question answering (Wu et al., 2017), many studies have been conducted to promote joint vision-and-language understanding. Visual grounding, which aims to find a specific region in an image given a query regarding an entity, is a fundamental task for enhancing the performance of various joint vision-and-language tasks (Plummer et al., 2015). For instance, in image captioning, it is important to ground to the corresponding image region while generating words for that region; in VQA, it is crucial to understand to which image region the question is referring. Because of the importance of visual grounding, many research efforts have been dedicated to improve its accuracy (Plummer et al., 2015; Wang et al., 2016a; Fukui et al., 2016; Rohrbach et al., 2016; Wang et al., 2016b; Yeh et al., 2017; Plummer et al., 2017; Chen et al., 2017; Yu et al., 2018b; Yang et al., 2020a,b; Dong et al., 2021).

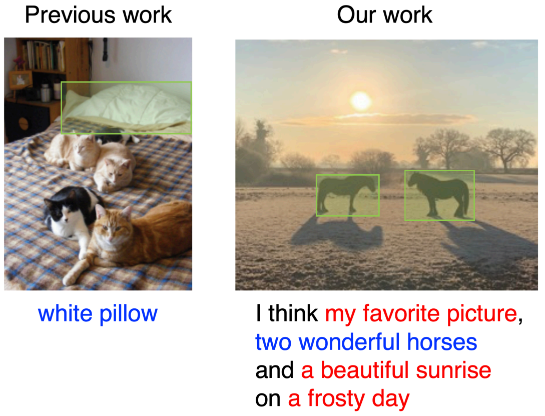


Figure 1: A comparison between previous visual grounding work and our flexible visual grounding work. In previous work, a query must be able to be grounded (see the left sub-figure), while our work can deal with both answerable and unanswerable visual grounding flexibly (in the right sub-figure, "two wonderful horses" can be grounded, while "my favorite picture," "a beautiful sunrise," and "a frosty day" cannot be grounded). The green bounding boxes are the ground-truth for answerable queries.

Previous visual grounding work assume that a query must be able to be grounded to an image region and create many datasets such as the Flickr30k entities (Plummer et al., 2015), RefClef (Kazemzadeh et al., 2014), RefCOCO, RefCOCO+ (Yu et al., 2016), RefCOCOg (Mao et al., 2016), and Visual7W datasets (Zhu et al., 2016) for the task. However, this assumption is not true in real-world multimedia data such as news, TV dramas,

[1] The social media dataset is available at https://github.com/ku-nlp/SMD4FVG.

Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics
Student Research Workshop, pages 285 - 299
May 22-27, 2022 ©2022 Association for Computational Linguistics

and social media, where entities in the text are not always able to be grounded to the visual data due to the fact that text and visual data in these multimedia data are unnecessarily directly corresponding to each other.

We name the case that a query can be grounded to an image region as *answerable* visual grounding; otherwise, *unanswerable* visual grounding from here. The ignorance of unanswerable visual grounding in previous work can lead to problems for downstream tasks. For instance, in VQA, if the VQA model cannot understand the case that entities in the question cannot be grounded to the image, it cannot deal with the case that a question cannot be answered given the image either. Therefore, a robust visual grounding model should be able to flexibly deal with both answerable and unanswerable visual grounding. In this work, we study this flexible visual grounding problem. Figure 1 compares our work with previous work.

To study flexible visual grounding, we construct two types of datasets. The first one is a pseudo dataset, which is constructed by randomly selecting queries from other images and combining it with a target image in the RefCOCO+ dataset (Yu et al., 2016). The second one is a social media dataset (SMD4FVG), which contains unanswerable real-world queries. We construct the SMD4FVG dataset by crawling tweets consisting of both images and text and annotating answerable and unanswerable queries via crowdsourcing.

Previous visual grounding models cannot handle unanswerable visual grounding. To give a model the ability to flexibly identify whether the input query can be grounded or not, we propose a novel method for unanswerable visual grounding by adding a pseudo region corresponding to a query that cannot be grounded. The model is then trained to ground to ground-truth regions for answerable queries and pseudo regions for unanswerable queries. Experiments conducted on both the pseudo and SMD4FVG datasets indicate that our model can flexibly process both answerable and unanswerable queries with high accuracy. In addition, we study the possibility of the usage of using the pseudo dataset to improve the accuracy on the SMD4FVG dataset.

The contributions of this paper are in three-folds:

- We propose a flexible visual grounding task that includes unanswerable visual grounding, where the unanswerable visual grounding problem has not been studied before.
- We construct a pseudo dataset based on the RefCOCO+ dataset and a social media dataset based on tweets consisting of both images and text via crowdsourcing for studying the flexible visual grounding task.
- We propose a flexible visual grounding model, which can deal with both answerable and unanswerable queries and achieves high accuracy on our datasets.

2 Related Work

Previous visual grounding studies have been conducted on different datasets. In the Flickr30k entities dataset (Plummer et al., 2015), a query corresponds to a noun phrase (i.e., entity) containing in a caption of an image. In the RefClef (Kazemzadeh et al., 2014), RefCOCO, RefCOCO+ (Yu et al., 2016), and RefCOCOg (Mao et al., 2016) datasets, a query is an phrase referring to an object in an image. In the Visual7W dataset (Zhu et al., 2016), a query corresponds to a question regarding an image region. However, all these datasets do not consider unanswerable visual grounding. In contrast, we propose flexible visual grounding and construct a pseudo dataset and a social media dataset.

Regarding visual grounding models, Plummer et al. (2015) proposed a method based on canonical correlation analysis (Hardoon et al., 2004) that learns joint embeddings of phrases and image regions. Wang et al. (2016a) proposed a two-branch neural network for joint phrasal and visual embeddings. Fukui et al. (2016) used multimodal compact bilinear pooling to fuse phrasal and visual embeddings. Rohrbach et al. (2016) proposed a method to first detect a candidate region for a given phrase and then reconstruct the phrase using the detected region. Wang et al. (2016b) proposed an agreement-based method, which encourages semantic relations among phrases to agree with visual relations among regions. Yeh et al. (2017) proposed a framework that can search over all possible regions instead of a fixed number of region proposals. Plummer et al. (2017) used spatial relationships between pairs of phrases connected by verbs or prepositions. Chen et al. (2017) proposed a reinforcement learning-based model that rewards the grounding results with image-level context. Yu et al. (2018b) improved the region proposal network by training it on the Visual Genome dataset

(Krishna et al., 2016) to increase the diversity of object classes and attribute labels. Sadhu et al. (2019) proposed to combine object detection and grounding models to deal with unseen nouns during training. Yang et al. (2020a) propagated relations among noun phrases in a query based on the linguistic structure of it. Yang et al. (2020b) addressed the long and complex queries by recursive sub-query construction. Dong et al. (2021) proposed a cross-lingual visual grounding task, which transfers the knowledge from an English model to improve the performance of a French model.

Inspired by the success of pre-training language models such as BERT (Devlin et al., 2019), vision-and-language pre-training on large image caption datasets such as the conceptual captions dataset (Sharma et al., 2018) has been proposed such as ViLBERT (Lu et al., 2019) VL-BERT (Su et al., 2020; Lu et al., 2020), and UNITER (Chen et al., 2020). Those vision-and-language pre-training models differ from the model architecture. Vision-and-language pre-training is evaluated on tasks including visual grounding. However, same to previous studies, the visual grounding task does not consider unanswerable cases (Lu et al., 2019; Su et al., 2020; Chen et al., 2020). Our flexible visual grounding model is based on the multi-task ViLBERT model (Lu et al., 2020), which achieves state-of-the-art performance on visual grounding.

3 Dataset Construction

Because there are no existing visual grounding datasets where unanswerable queries are contained, we present two ways to construct two types of datasets to study the flexible visual grounding problem.

3.1 RefCOCO+ Pseudo Dataset

As the construction of a new large-scale dataset is costive and time-consuming, firstly, we constructed a pseudo dataset based on the RefCOCO+ dataset (Yu et al., 2016) using the negative pair sampling method presented in (Yu et al., 2018a). To generate unanswerable data, we randomly select an image and a query of another image from the RefCOCO+ dataset and combine them as a pair of visual grounding data. Because the query is from a different image, we can assume that the query cannot be grounded to the selected image. However, there is still a possibility that the randomly selected query can be grounded to the image, which may lead to noise. We will discuss this problem in Section 6.1. Next, we combined the generated unanswerable data to the original RefCOCO+ dataset to make a pseudo dataset containing both answerable and pseudo unanswerable queries.

3.2 Social Media Dataset (SMD4FVG)

Unanswerable visual grounding exists in real-world multimedia data consisting of both text and visual information such as news, TV dramas, and social media. Among these, social media is one typical case where there are many unanswerable visual grounding data because the text and visual information posted by users are not necessarily closely related to each other. Due to this characteristic, in social media, there could be more unanswerable visual grounding data than answerable ones. This might result in an unbalanced dataset, making training and evaluation difficult. In order to construct a balanced dataset, we propose a pipeline shown in Figure 2. We describe each step in detail in this section.

Data Crawling

To construct the SMD4FVG dataset, we first crawled image and text pairs from Twitter. We will follow the fair use policy of Twitter regarding copyright of the crawled data.[2] We used Twitter's official library tweepy[3] for this process. In order to inherit previous visual grounding studies, we decided to crawl data from the same domain as the RefCOCO+ dataset. To this end, we searched the hashtags in Twitter that match the object classes in the RefCOCO+ dataset and only crawled the data that hit. As a result, $20,941$ tweets of images and text pairs were crawled.

Image Filtering

In order to construct a visual grounding dataset balanced on both answerable and unanswerable queries, we further conducted image filtering from the crawled tweets. For the image filtering process, we used EfficientnNet (Tan and Le, 2019) to classify images, Yolov4 (Bochkovskiy et al., 2020) to detect objects and CRAFT (Baek et al., 2019) to detect text in images.

The EfficientNet model was pre-trained on the ImageNet dataset (Deng et al., 2009). With the

[2] https://help.twitter.com/en/rules-and-policies/fair-use-policy

[3] https://www.tweepy.org/

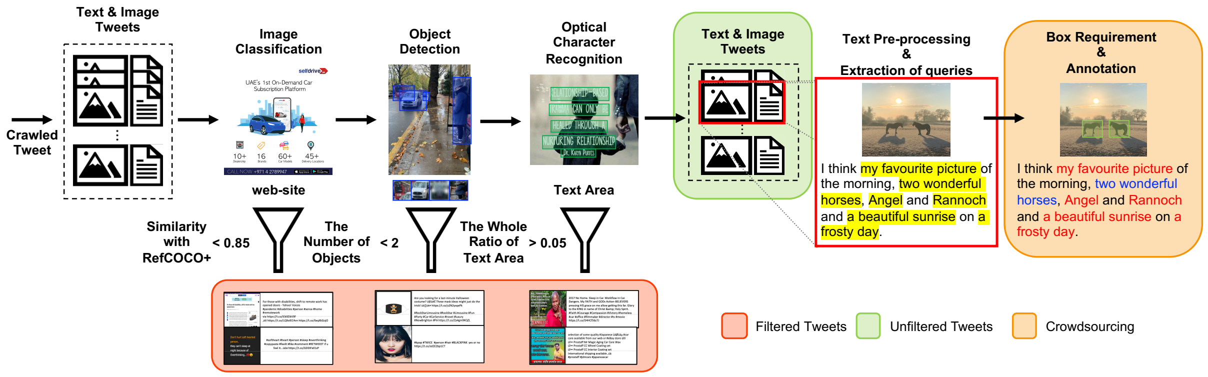

Text & Image Tweets
Crawled Tweet
Image Classification
web-site
Object Detection
Optical Character Recognition
Text Area
Similarity with RefCOCO+
< 0.85
The Number of Objects
< 2
The Whole Ratio of Text Area
> 0.05
Text & Image Tweets
Text Pre-processing & Extraction of queries
I think my favourite picture of the morning, two wonderful horses, Angel and Rannoch and a beautiful sunrise on a frosty day.
Box Requirement & Annotation
I think my favourite picture of the morning, two wonderful horses, Angel and Rannoch and a beautiful sunrise on a frosty day.
Filtered Tweets
Unfiltered Tweets
Crowdsourcing

Figure 2: The pipeline for constructing the social media dataset. After crawling tweets containing both images and text, we first filter images that do not belong to the RefCOCO+ classes, contain less than two objects, or are dominated with text in the image step by step. After that, we extract noun phrases as queries in the tweet text. Finally, we annotate answerable and unanswerable queries via crowdsourcing in two steps where in the first step, unanswerable queries are identified, and in the second step, bounding boxes are annotated for answerable queries.

same purpose of inheriting previous visual grounding studies, from the ImageNet classes output by EfficientNet, we only chose the classes similar to RefCOCO+ classes and removed the others. When determining the similarities between the RefCOCO+ classes, we calculated the Wu & Palmer similarity (Wu and Palmer, 1994) and chose classes that surpassed a similarity score of 0.85. It calculates similarity by considering the depths of the two synsets $(s1, s2)$ within the WordNet (Feinerer and Hornik, 2020) hierarchy, along with the depth of the least common subsumer (LCS) as:

$$\mathrm{Wu-Palmer} = 2 * \frac{depth(LCS(s1, s2))}{depth(s1) + depth(s2)} \tag{1}$$

As a result of the image classification-based filtering, the crawled 20,941 tweets decreased to 6,813 tweets.

For the next step, we filtered more tweets using the Yolov4 object detection model. The object detection model was pre-trained with the Microsoft COCO dataset (Lin et al., 2014). We chose images that had two or more objects because images with only one single object or background are considered to be too easy for our task. As a result, 4,028 tweets were chosen from the 6,813 tweets.

In the crawled tweets, we found that many images consisted of mostly text and website information. As visual grounding is almost impossible for text/website-dominated images, we further filtered those images. To this end, we used the optical character recognition model of CRAFT. Based on the results of the optical character recognition model, we calculated a text proportion ratio in an image. We only kept images that had a proportion ratio lower than 0.05 with respective to the entire image. As a result, 3,425 images were left.

Due to the limitations of the above image processing models, advertisement, inappropriate, and duplicate images were still left in the dataset after the above filtering process. Therefore, we further manually checked the data and discarded them. As a result, 988 tweets were finally left.

Query Extraction

Tweets contain emoji, links, and mentions, which make query extraction difficult. Therefore, we preprocessed the data and eliminated those expressions. From the pre-processed text, we extracted sentences and used the chunking model (Akbik et al., 2018) to chunk the noun phrases within the sentences. We did not use the pronoun (such as he, her, she) and relative pronoun (such as which, who, that) as queries. As for complex noun phrases that contain other noun phrases within them, we split them and only used single noun phrases as queries. As a result, we obtained 8,827 queries for the 988 images.

Crowdsourcing Annotation

From the 8,827 pairs of image and query obtained, we annotated image regions that can be grounded by queries and finally constructed the SMD4FVG dataset. For the annotation, we used Amazon Mechanical Turk. The compensation was 8-9 dollars per hour.

The annotation process consists of two steps.[4] The first step is the "bounding box requirement" task. In this step, we asked workers if a query can be grounded, and if not, which of the following cases it belongs to: 1) What the query refers to cannot be seen in the image. 2) The query does not refer to something specific in the image but rather to the background. 3) The query is an abstract noun that might be confusing based on the contents of the image.

In case 1, the query refers to an entity, but the image does not contain that entity. For instance, in the right part of Figure 1, the query "my favorite picture" entity does not appear in the image. In case 2, if the query is the background of an image, it might make the annotation regions different by different workers, or as there are many objects in the background, it might make the definition of background vague. For instance, in the right part of Figure 1, it is hard to clearly determine the region for the query "a beautiful sunrise." Also, there might be many objects in the annotation. Therefore, we asked workers to annotate this case as unanswerable. In case 3, if the query is an abstract noun, the judgment of annotation might differ from workers. For instance, if the query is "sport," and some workers might define "sport" as a person doing a sport and determine the query as answerable based on the contents of an image, and some workers might define "sport" as something invisible and determine the query as unanswerable. Thus, we set this case as unanswerable. As a result of the crowdsourcing annotation for this step, we obtained $6,941$ unanswerable queries in total.

The second step is the "drawing the bounding box" task. In this step, the annotation was done for data that were not annotated as unanswerable in the first step. Workers were asked to draw a bounding box for an image region corresponding to a query. The difficult part of this process was when there were multiple instances that corresponded to one query in an image. In this case, we instructed the workers to annotate multiple instances to one bounding box if the instances are not clearly separated; otherwise, we annotate them with individual bounding boxes. Besides that, queries in social media data can contain proper nouns, which are special compared to previous datasets and could be interesting to study; thus, we asked workers to indicate if an answerable query belongs to these. In total, $1,886$ answerable queries were annotated, among which 576 queries belong to proper nouns.

Finally, we manually checked the results of the two steps. We checked 100 unanswerable pairs and found that 7 of them were wrongly labeled. Most of them were simple misses where the entity that the query refers to does exist in an image, which we plan to improve as our future work. In addition, we checked and corrected the bounding boxes that were miss-labeled by workers of all answerable pairs. As a result, we obtained $8,827$ annotated query and image pairs for our SMD4FVG dataset.

4 Flexible Visual Grounding Model

We propose to add a pseudo region to a visual grounding model to achieve flexible visual grounding for both answerable and unanswerable queries. An overview of our proposed model is shown in Figure 3. In this section, we first present our visual grounding model, followed by the way to add pseudo regions for unanswerable queries.

4.1 Visual Grounding Model

Our visual grounding model follows (Lu et al., 2020), which consists of 2 stages. In the first stage, we extract region proposals and feature vectors of all regions with an object detection model. We employ the Faster RCNN (Ren et al., 2015) model in the first stage. In the second stage, a similarity score between a region proposal and an input query is calculated. We utilize the multi-task ViLBERT (Lu et al., 2020) for the calculation of the similarity between a region proposal and the input query. Our model is trained to minimize a binary cross-entropy (BCE) loss between a label vector and a similarity score vector similar to (Sadhu et al., 2019). In inference, the input query will be grounded to the region with the highest similarity score.

In detail, after extracting a feature vector $\mathbf{f}_v \in \mathbb{R}^{d_v}$ for a region proposal by Faster RCNN, a spatial vector $\mathbf{f}_s \in \mathbb{R}^5$ is incorporated to it. The spatial vector is encoded to a 5-d vector from normalized top-left and bottom-right coordinates as:

$$\mathbf{f_s} = \left[\frac{x_{tl}}{W}, \frac{y_{tl}}{H}, \frac{x_{br}}{W}, \frac{y_{br}}{W}, \frac{wh}{WH}\right], \quad (2)$$

where (x_{tl}, y_{tl}) is the top-left coordinate, (w_{br}, y_{br}) is the bottom-right coordinate, w and h are the the width and the height of the region, and W and H are the width and the height of the image, respectively. The spatial vector is then projected to match

[4] The screenshot of the interfaces for these two steps can be found in Appendix A.

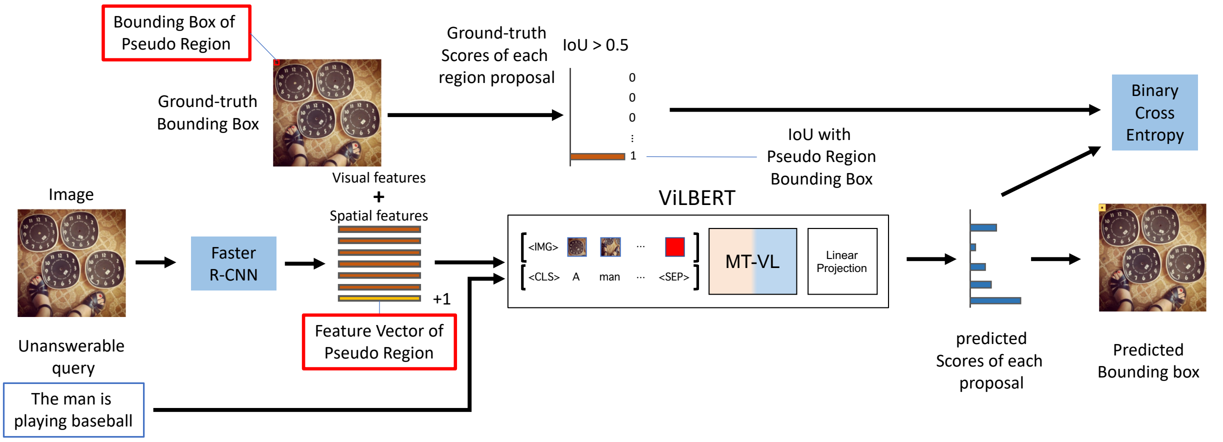

Bounding Box of Pseudo Region
Ground-truth Bounding Box
Ground-truth Scores of each region proposal
IoU > 0.5
0
0
0
⋮
1
IoU with Pseudo Region Bounding Box
Binary Cross Entropy
Image
Faster R-CNN
Visual features + Spatial features
+1
Feature Vector of Pseudo Region
ViLBERT
<IMG> ...
<CLS> A man ... <SEP>
MT-VL
Linear Projection
predicted Scores of each proposal
Predicted Bounding box
Unanswerable query
The man is playing baseball

Figure 3: The proposed flexible visual grounding model. For an unanswerable query, we add a pseudo region and train the model to ground the query to the pseudo region.

the dimension of the visual feature by a learnable weight matrix $W_s \in \mathbb{R}^{5 \times d_v}$ and then added to $\mathbf{f}_v$ to generate the final region feature vector $\mathbf{v}_r$ as:

$$\mathbf{v}_r = \mathbf{f}_v + W_s \mathbf{f}_s. \tag{3}$$

The query is given in both training and inference. It is denoted as $\mathbf{q}$. Next, $\mathbf{v}_r$ and $\mathbf{q}$ are input to the multi-task ViLBERT model, which generates a representation $\mathbf{h}_i \in \mathbb{R}^{d_i}$ for the ith region and the query as:

$$\mathbf{h}_i = \texttt{ViLBERT}(\mathbf{v}_r, \mathbf{q}). \tag{4}$$

$\mathbf{h}_i$ is then used to calculate a similarity score for the ith region by:

$$s_i = W_i \mathbf{h}_i, \tag{5}$$

where $W_i \in \mathbb{R}^{d_i \times 1}$ is a learnable weight matrix.

The ground-truth label score is set to 1 if the IoU between a region proposal and the ground-truth region is larger than 0.5; otherwise, it is set to 0. The similarity score vector s_{ji} and the ground-truth label vector l_{ji} for the ith region in the jth image are then used to minimize a BCE loss as:

$$\texttt{BCE} = -\frac{1}{N} \sum_{j=1}^{N} \sum_{i=1}^{M} l_{ji} log(s_{ji}) + (1 - l_{ji}) log(1 - s_{ji}), \tag{6}$$

where N is the number of image and query pairs in a dataset, and M is the number of region proposals for an image.

4.2 Pseudo Region

To make our visual grounding model deal with unanswerable queries, we propose to incorporate a pseudo region corresponding to an unanswerable query into the region proposals. An example is shown in Figure 3. In Figure 3, the input query “man is playing baseball" is not related to the input image, where the image is about feet and clocks; thus, the query cannot be grounded to the image. For this query, we add a pseudo region to the regions proposed by Faster RCNN (Ren et al., 2015). The position of the pseudo region is set to the top-left of the input image, and all the x and y coordinate values of its spatial vector are set to 0 in Eq. (2). All components of the feature vector $\mathbf{f}_v \in \mathbb{R}^{d_v}$ for the pseudo region are set to $+1$.

Our visual grounding model calculates the similarity score between the pseudo region incorporated region vectors and the query same as Section 4.1. The model is then trained to give the highest similarity score for the pseudo region when the query cannot be grounded. During inference, the model will output the region with the highest score as the prediction. For instance, in the example of Figure 3, the pseudo region will be chosen for the input query because the input query is not corresponding to the input image.

5 Experimental Settings

In our experiments, we verify the effectiveness of the proposed model on both the RefCOCO+ pseudo and SMD4FVG datasets. Here, we first describe the statistics of each dataset and settings, followed by training details.

5.1 Settings on the RefCOCO+ Pseudo Dataset

For the pseudo dataset, based on the RefCOCO+ dataset, we generated unanswerable data and com-

Dataset	Split	Answerable	Unanswerable
Pseudo	Train	42,278	21,139
	Validation	3,805	1,905
	Test	3,773	1,886
SMD4FVG	Train	1,270	4,775
	Validation	330	1,097
	Test	286	1,069

Table 1: Statistics of our datasets (i.e., number of query and image pairs).

bined them with the original dataset with the ratio of 1:2. The upper part of Table 1 shows the statistics of the pseudo dataset.

For the pseudo dataset, we investigated the performance of our model with the following settings:

- **RefCOCO+**: A baseline that trained our visual grounding model in Section 4 on the original RefCOCO+ dataset to evaluate answerable visual grounding only, and compared the performance with (Lu et al., 2020).

- **RefCOCO+Thres**: A baseline based on the RefCOCO+ setting but sets a threshold according to the similarity score (Eq. (4)) distribution for all queries during inference. Queries with the highest similarity scores below the threshold were treated as unanswerable otherwise answerable. The threshold was tuned on the validation split of the pseudo dataset to achieve the highest accuracy for all queries.

- **Pseudo**: We directly trained and evaluated our model on the pseudo dataset.

- **SM→Pseudo**: We first trained our model on the training data of the SMD4FVG dataset and then further fine-tuned it on the pseudo dataset. We hope that the annotated SMD4FVG dataset could boost the performance on the pseudo dataset.

5.2 Settings on the SMD4FVG Dataset

The lower part of Table 1 shows the statistics of the SMD4FVG dataset, where we split the annotated $8,827$ query and image pairs into train/validation/test with a 69%:16%:15% distribution. We evaluated the performance on the SMD4FVG dataset with the following settings:

- **RefCOCO+Thres**: A baseline similar to the RefCOCO+Thres setting on the pseudo dataset, but the threshold was tuned on the validation split of the SMD4FVG dataset.

- **Pseudo**: Aiming to investigate the difference between the pseudo and SMD4FVG datasets, we trained our model on the training data of the pseudo dataset and evaluated it on the SMD4FVG dataset.

- **SM**: This is a straightforward setting that directly trained and evaluated our visual grounding model on the SMD4FVG dataset.

- **Pseudo→SM**: We first trained our model on the training data of the pseudo dataset and then further fine-tuned it on the SMD4FVG dataset. We hope that the large scale of the pseudo dataset could boost the performance on the SMD4FVG dataset.

5.3 Training Details

Visual features and region proposals were extracted from the ResNeXT-152 Faster-RCNN model (Ren et al., 2015) trained on the Visual Genome dataset (Krishna et al., 2016) with an attribute loss. It was not fine-tuned during training. We used the multi-task ViLBERT model (Lu et al., 2020) for calculating the similarity score between region proposals and the query, which contains a 6 / 12 layer of transformer blocks for visual/linguistic streams individually. The multi-task ViLBERT was trained simultaneously with 4 vision-and-language tasks on 12 datasets. We set the region feature dimension d_v to $2,048$, the joint ViLBERT representation dimension d_i to $1,024$, and the number of region proposals N to 100. We trained our model on 8 TitanX GPUs with a batch size of 256, 20 epochs, and the AdamW optimizer with a linear warmup and linear decay learning rate scheduler following (Lu et al., 2020) for all settings.

6 Results

6.1 Results on the Pseudo Dataset

The upper part of Table 2 shows the accuracy of our model on the pseudo dataset. For the RefCOCO+ setting, our model achieves an accuracy of 73.3%, which is almost the same as the result 73.2% when we evaluated the original model of (Lu et al., 2020) using their codes. This indicates that adding a pseudo region has little effect on the performance for answerable visual grounding. However, it cannot deal with unanswerable queries due to the absence of such data in the RefCOCO+ dataset. The

Dataset	Setting	Ans.	Unans.	All
Pseudo	RefCOCO+	73.3	N/A	73.3
	RefCOCO+Thres	**90.3**	46.9	75.9
	Pseudo	69.7	**91.2**	76.8
	SM→Pseudo	70.3	89.9	**76.9**
SMD4FVG	RefCOCO+Thres	0	**100.0**	78.9
	Pseudo	**49.7**	65.6	62.2
	SM	31.8	95.0	**81.7**
	Pseudo→SM	41.3	91.3	80.7

Table 2: Visual grounding results on the pseudo and SMD4FVG datasets. Ans., Unans., and All denote the accuracy for answerable, unanswerable, and all queries, respectively.

RefCOCO+Thres setting works well for answerable queries but fails for answerable ones. The similarity score distribution is in Appendix B.

For the pseudo setting, our model achieves an accuracy of 69.7% and 91.2% for answerable and unanswerable queries, respectively. Our model can ground unanswerable queries with high accuracy. However, it drops 2.6% point for answerable queries compared to the RefCOCO+ setting. We think the reason for this is due to the mixture of unanswerable queries to the original RefCOCO+ dataset, leading the judgment to answerable visual grounding be more complex. SM→Pseudo only slightly boots the All accuracy due to the small-scale of the SMD4FVG dataset. Some incorrect predictions for unanswerable queries are due to the randomness of the dataset, and qualitative examples can be found in Appendix C.

6.2 Results on the SMD4FVG Dataset

The lower part of Table 2 shows the accuracy of our model on the SMD4FVG dataset. We can see that the RefCOCO+Thres setting forces all queries to be unanswerable ones. The similarity score distribution can be found in Appendix B.

Among the other three settings, the pseudo setting achieves the highest accuracy of 49.7% for answerable queries. We think the reason for this is that there are only a few answerable queries in the SMD4FVG dataset, while both the amount and ratio for that are higher in the pseudo dataset, making the model learn answerable grounding well. However, the accuracy for unanswerable queries is only 65.6%, which is significantly worse than the other two settings that use the SMD4FVG dataset for training. We think this is due to the different characteristics of unanswerable queries in the pseudo and SMD4FVG datasets, wherein the pseudo dataset the unanswerable queries are unrelated to the images, but in the SMD4FVG dataset they are more complex. The SM setting achieves high accuracy of 95.0% for unanswerable queries and the best accuracy of 81.7% for all queries. The reason for this can be that our model is optimized in the SMD4FVG dataset directly with the SM setting. However, the accuracy for answerable queries with the SM setting is the lowest due to the small ratio of answerable queries and complex answerable queries in the SMD4FVG dataset. The Pseudo→SM setting achieves a trade-off between the pseudo and SM settings, where there is an improvement for answerable queries compared to the SM setting and a big improvement for unanswerable queries compared to the pseudo setting. We think the reason for this is that Pseudo→SM can take the balance between the pseudo and SM settings via fine-tuning the model pre-trained on the pseudo dataset to the SMD4FVG dataset. We also observe a 1% accuracy drop of all queries from SM to Pseudo→SM. We think it is caused by the big ratio of unanswerable queries in the SMD4FVG dataset. The SM model was more biased to unanswerable queries and thus performed better in accuracy for all queries because of the big ratio of unanswerable queries. Qualitative examples can be found in Appendix C.

For both the pseudo and SMD4FVG datasets, we observe better performance on unanswerable queries than answerable queries besides RefCOCO+Thres on the pseudo dataset. We think the reason could be that it is much easier to learn that a query is unrelated to an image (i.e., unanswerable) instead of finding the exact region that a query refers to (i.e., answerable) by our models.

7 Conclusion

Previous studies on visual grounding ignored the case of unanswerable queries, which is common in real-world such as social media data. In this paper, we proposed flexible visual grounding to address both answerable and unanswerable visual grounding. To this end, we constructed a pseudo dataset based on the RefCOCO+ dataset and a social media dataset based on tweets consisting of both images and text via crowdsourcing. In addition, we proposed a flexible visual grounding model, which can deal with both answerable and unanswerable queries. Experiments on our datasets indicated that our model could achieve high accuracy, especially for unanswerable queries, but there is still room for further improvement.

To make our social media dataset balanced, we constrained it to the RefCOCO+ classes, which may also limit the ability of our model on real-world data. In the future, we plan to construct a dataset without such constraints.

Acknowledgement

This work was supported by ACT-I, JST.

References

Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. Contextual string embeddings for sequence labeling. In *COLING 2018, 27th International Conference on Computational Linguistics*, pages 1638–1649.

Youngmin Baek, Bado Lee, Dongyoon Han, Sangdoo Yun, and Hwalsuk Lee. 2019. Character region awareness for text detection. *CoRR*, abs/1904.01941.

Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. 2020. YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv e-prints*, page arXiv:2004.10934.

Kan Chen, Rama Kovvuri, and Ram Nevatia. 2017. Query-guided regression network with context policy for phrase grounding. In *ICCV*, pages 824–832.

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

J. Deng, W. Dong, R. Socher, L. Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL: HLT*, pages 4171–4186.

Wenjian Dong, Mayu Otani, Noa Garcia, Yuta Nakashima, and Chenhui Chu. 2021. Cross-lingual visual grounding. *IEEE Access*, 9:349–358.

Ingo Feinerer and Kurt Hornik. 2020. *wordnet: WordNet Interface*. R package version 0.1-15.

Akira Fukui, Dong Huk Park, Daylen Yang, Anna Rohrbach, Trevor Darrell, and Marcus Rohrbach. 2016. Multimodal compact bilinear pooling for visual question answering and visual grounding. In *EMNLP*, pages 457–468.

David R. Hardoon, Sandor R. Szedmak, and John R. Shawe-taylor. 2004. Canonical correlation analysis: An overview with application to learning methods. *Neural Computation*, 16(12):2639–2664.

Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. 2014. ReferItGame: Referring to objects in photographs of natural scenes. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 787–798, Doha, Qatar. Association for Computational Linguistics.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, Michael Bernstein, and Li Fei-Fei. 2016. Visual genome: Connecting language and vision using crowdsourced dense image annotations.

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 *ECCV*, pages 740–755.

Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *NeurIPS*, pages 13–23.

Jiasen Lu, Vedanuj Goswami, Marcus Rohrbach, Devi Parikh, and Stefan Lee. 2020. 12-in-1: Multi-task vision and language representation learning. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L. Yuille, and Kevin Murphy. 2016. Generation and comprehension of unambiguous object descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Bryan A. Plummer, Arun Mallya, Christopher M. Cervantes, Julia Hockenmaier, and Svetlana Lazebnik. 2017. Phrase localization and visual relationship detection with comprehensive image-language cues. In *ICCV*, pages 1928–1937.

Bryan A. Plummer, Liwei Wang, Chris M. Cervantes, Juan C. Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. 2015. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *ICCV*, pages 2641–2649.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, pages 91–99.

Anna Rohrbach, Marcus Rohrbach, Ronghang Hu, Trevor Darrell, and Bernt Schiele. 2016. Grounding of textual phrases in images by reconstruction. In *ECCV*, pages 817–834.

Arka Sadhu, Kan Chen, and Ram Nevatia. 2019. Zero-shot grounding of objects from natural language queries. In *The IEEE International Conference on Computer Vision (ICCV)*.

Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. 2018. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *ACL*, pages 2556–2565.

Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. 2020. Vl-bert: Pre-training of generic visual-linguistic representations. In *ICLR*.

Mingxing Tan and Quoc V. Le. 2019. Efficientnet: Rethinking model scaling for convolutional neural networks. *CoRR*, abs/1905.11946.

Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. 2015. Show and tell: A neural image caption generator. In *CVPR*, pages 3156–3164.

Liwei Wang, Yin Li, and Svetlana Lazebnik. 2016a. Learning deep structure-preserving image-text embeddings. In *CVPR*, pages 5005–5013.

Mingzhe Wang, Mahmoud Azab, Noriyuki Kojima, Rada Mihalcea, and Jia Deng. 2016b. Structured matching for phrase localization. In *ECCV*, pages 696–711.

Qi Wu, Damien Teney, Peng Wang, Chunhua Shen, Anthony Dick, and Anton van den Hengel. 2017. Visual question answering: A survey of methods and datasets. *CVIU*, pages 1–20.

Zhibiao Wu and Martha Palmer. 1994. Verb semantics and lexical selection. *CoRR*, abs/cmp-lg/9406033.

Sibei Yang, Guanbin Li, and Yizhou Yu. 2020a. Propagating over phrase relations for one-stage visual grounding. In *ECCV*.

Zhengyuan Yang, Tianlang Chen, Liwei Wang, and Jiebo Luo. 2020b. Improving one-stage visual grounding by recursive sub-query construction. In *ECCV*.

Raymond Yeh, Jinjun Xiong, Wen-Mei W. Hwu, Minh Do, and Alexander G. Schwing. 2017. Interpretable and globally optimal prediction for textual grounding using image concepts. In *NIPS*, pages 1909–1919.

Licheng Yu, Zhe Lin, Xiaohui Shen, Jimei Yang, Xin Lu, Mohit Bansal, and Tamara L Berg. 2018a. Mattnet: Modular attention network for referring expression comprehension. In *CVPR*.

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

Zhou Yu, Jun Yu, Chenchao Xiang, Zhou Zhao, Qi Tian, and Dacheng Tao. 2018b. Rethinking diversified and discriminative proposal generation for visual grounding. In *IJCAI*, pages 1114–1120.

Yuke Zhu, Oliver Groth, Michael Bernstein, and Li Fei-Fei. 2016. Visual7w: Grounded question answering in images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

A Annotation Interfaces

Figure 4 shows the screenshot of the first step of crowdsourcing. This step is the "bounding box requirement" task. We instruct workers to check if the given query is answerable or not. For unanswerable queries, we further ask workers to check which unanswerable type the query is.

Figure 5 shows the screenshot of the second step of crowdsourcing. This step is the "drawing bounding box" task. For an answerable query, we instruct workers to draw bounding boxes to which the query refers.

B Similarity Score Distribution

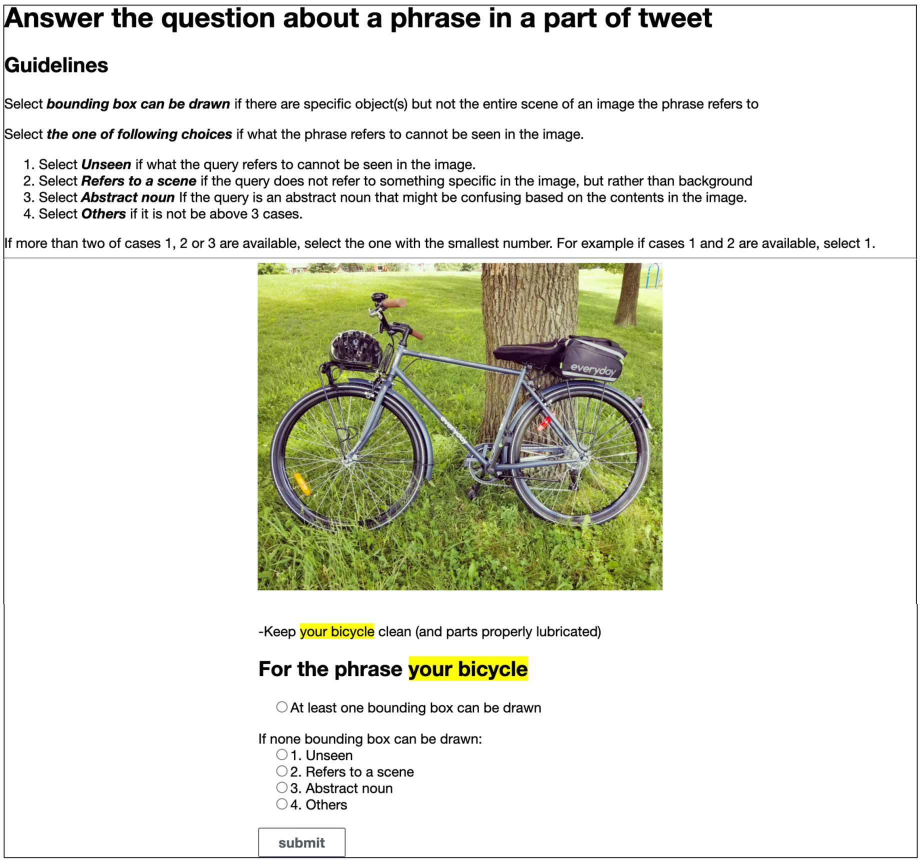
Figure 6 shows the similarity score distribution of the RefCOCO+Thres setting on the testsets of the pseudo dataset and SMD4FVG dataset, respectively. We can see that the similarity score and the grounding possibility have a very low correlation.

C Qualitative Examples

Figure 7 shows examples of our model with the RefCOCO+ setting on unanswerable queries in the pseudo dataset. We can see that the RefCOCO+ setting cannot identify unanswerable queries, which gives wrong predictions for them. However, there are also some ambiguous queries, such as the ones in examples 1, 6, and 7, for which we cannot confidently claim that the predictions are wrong due to the random combination characteristics of unanswerable queries in the pseudo dataset.

Figure 8 shows example outputs of our model with the pseudo setting. Examples 1 and 2 in Figure 8 are two successful examples for answerable visual grounding; we can see that our model can ground queries with and without modifiers. Examples 3 and 4 in Figure 8 are two successful examples for unanswerable visual grounding; we can see that for the queries that are unrelated to the images, our model can correctly identify that they cannot be grounded. Examples 5 and 6 in Figure 8 are two unsuccessful examples for answerable visual grounding; our model fails on example 5 in Figure 8 where the ground-truth is the other person with the number 160 on the vest; for example 6 in Figure 8, the query "taller one" itself is actually ambiguous, and our model makes the judgment that it cannot be grounded, while the ground-truth is annotated for the "taller refrigerator" in the RefCOCO+ dataset. Although our model achieves 91.2% accuracy for unanswerable queries, it still makes some mistakes. Examples 7 and 8 in Figure 8 show two unsuccessful examples for unanswerable visual grounding; we can see that for example 7 in Figure 8, the query "lady" actually can be grounded, but it is annotated as unanswerable in our pseudo dataset due to the fact that the query is taken from another image randomly and it could be grounded in coincidence; the query for example 8 in Figure 8 is again ambiguous, and thus it is actually difficult to claim that our model is wrong here.

Figure 9 shows example outputs of our model with the SNS setting, which achieves the best overall accuracy among the three settings. Examples 1 and 2 in Figure 9 are two successful examples for answerable visual grounding; we can see that our model can do grounding for both a single object (example 1) and multiple objects (example 2). Examples 3 and 4 in Figure 9 are two successful examples for unanswerable visual grounding; we can see that our model correctly identifies that the abstract noun query "sport" and the query "the east coast" that cannot be inferred from the image directly, cannot be grounded. Examples 5 and 6 in Figure 9 are two unsuccessful examples for answerable visual grounding; for example 5, the query "airbus320ceo" is a proper noun, which is difficult for grounding; while for example 6, "coach" is difficult to infer from the image though "bus" is clear. Examples 7 and 8 in Figure 9 show two unsuccessful examples for unanswerable visual grounding; for example 7, due to the failure of our query extraction model, an adjective query "automotive" is generated, which should not be grounded; for example 8, it is a human dressed up as a bear but not a real bear, and thus should not be grounded.



Answer the question about a phrase in a part of tweet

Guidelines

Select ***bounding box can be drawn*** if there are specific object(s) but not the entire scene of an image the phrase refers to

Select ***the one of following choices*** if what the phrase refers to cannot be seen in the image.

1. Select ***Unseen*** if what the query refers to cannot be seen in the image.
2. Select ***Refers to a scene*** if the query does not refer to something specific in the image, but rather than background
3. Select ***Abstract noun*** If the query is an abstract noun that might be confusing based on the contents in the image.
4. Select ***Others*** if it is not be above 3 cases.

If more than two of cases 1, 2 or 3 are available, select the one with the smallest number. For example if cases 1 and 2 are available, select 1.

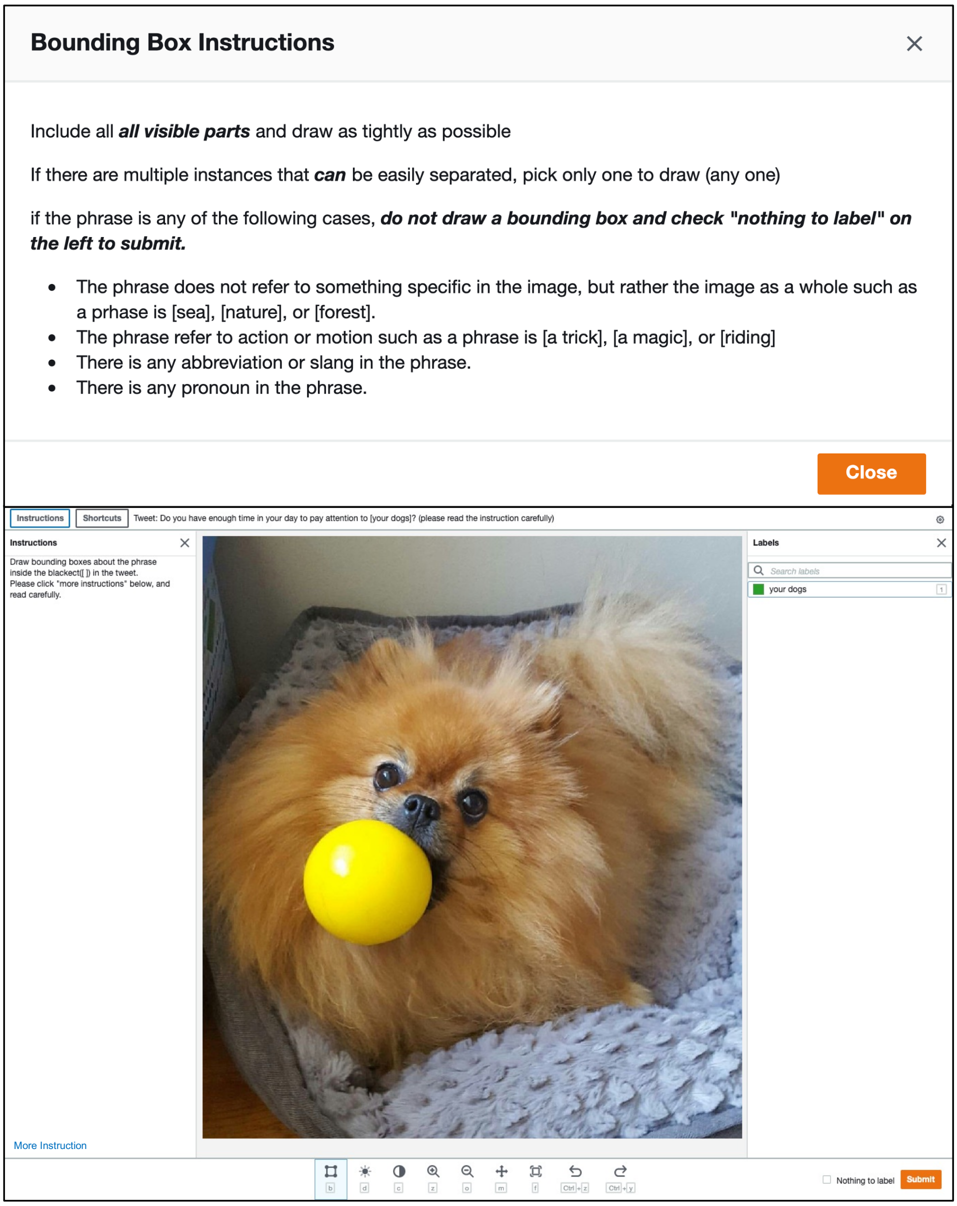
-Keep your bicycle clean (and parts properly lubricated)

For the phrase your bicycle

○ At least one bounding box can be drawn

If none bounding box can be drawn:
○ 1. Unseen
○ 2. Refers to a scene
○ 3. Abstract noun
○ 4. Others

Figure 4: The bounding box requirement interface. This is the first step of crowdsourcing. In this step, we instruct workers to check whether the given query is answerable or not. If the query is unanswerable, we ask workers to further check which unanswerable type the query is.



Bounding Box Instructions

Include all ***all visible parts*** and draw as tightly as possible

If there are multiple instances that ***can*** be easily separated, pick only one to draw (any one)

if the phrase is any of the following cases, ***do not draw a bounding box and check "nothing to label" on the left to submit.***

- The phrase does not refer to something specific in the image, but rather the image as a whole such as a prhase is [sea], [nature], or [forest].
- The phrase refer to action or motion such as a phrase is [a trick], [a magic], or [riding]
- There is any abbreviation or slang in the phrase.
- There is any pronoun in the phrase.

Tweet: Do you have enough time in your day to pay attention to [your dogs]? (please read the instruction carefully)

Instructions

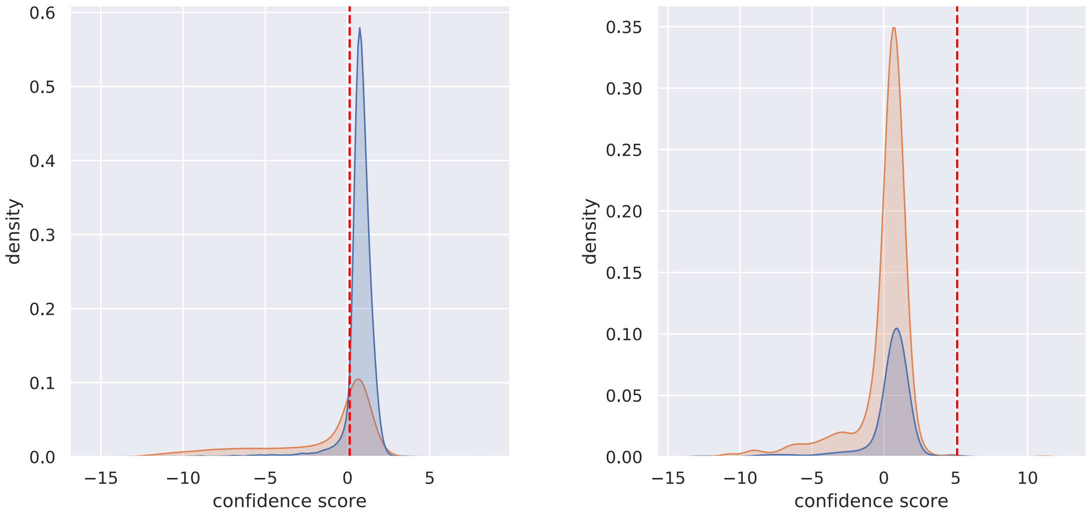
Draw bounding boxes about the phrase inside the blackect([ ]) in the tweet.
Please click "more instructions" below, and read carefully.

Labels

🟩 your dogs 1

□ Nothing to label

Figure 5: The drawing bounding box interface. This is the second step of crowdsourcing. In this step, we instruct workers to draw bounding boxes to which the query refers. The annotation is done for query and image pairs that are classified as answerable in the first step.



Pseudo dataset

SMD4FVG dataset

Figure 6: The similarity score distribution of the RefCOCO+Thres setting on the testsets of the pseudo dataset and SMD4FVG dataset, respectively. X-axis and Y-axis denote the similarity/confidence score and density, respectively. The solid blue and orange curves represent answerable and unanswerable queries, respectively. The vertical dotted red lines denote the thresholds.

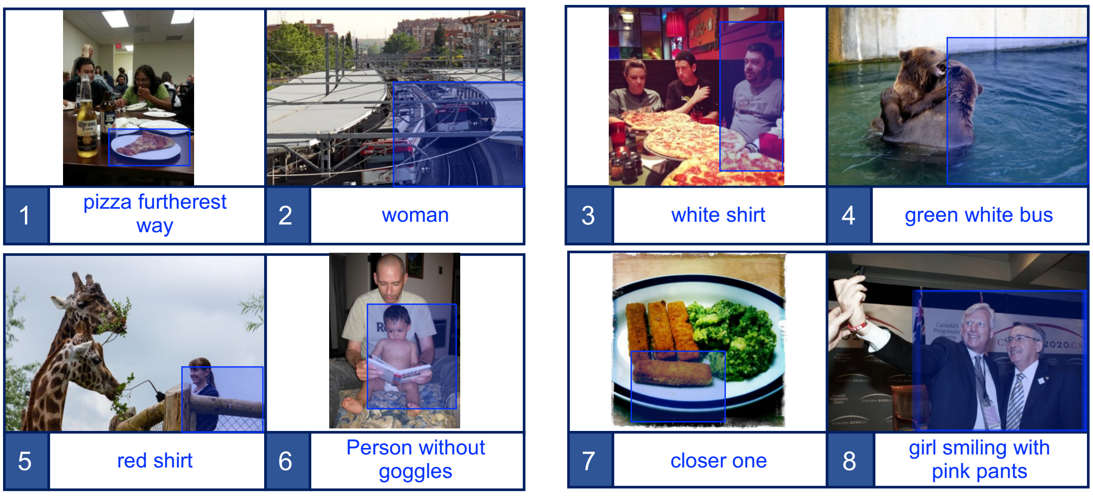


Figure 7: Examples of visual grounding for unanswerable queries in the pseudo dataset. The blue bounding boxes are the prediction of our model with the RefCOCO+ setting.

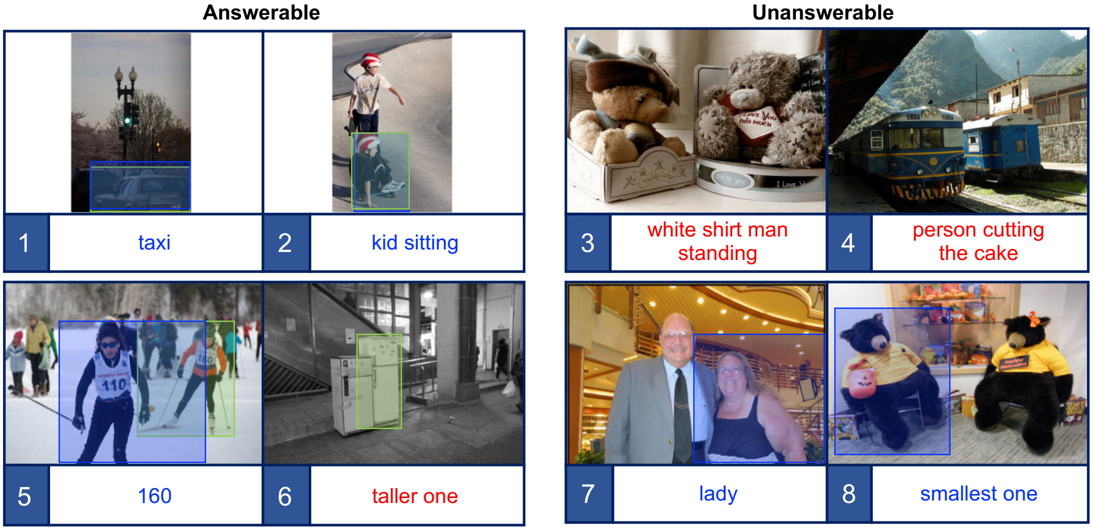


Figure 8: Examples of successful (top) and unsuccessful (bottom) visual grounding for answerable and unanswerable queries in the pseudo dataset. The green and blue bounding boxes are ground-truth and the prediction of our model with the pseudo setting, respectively.

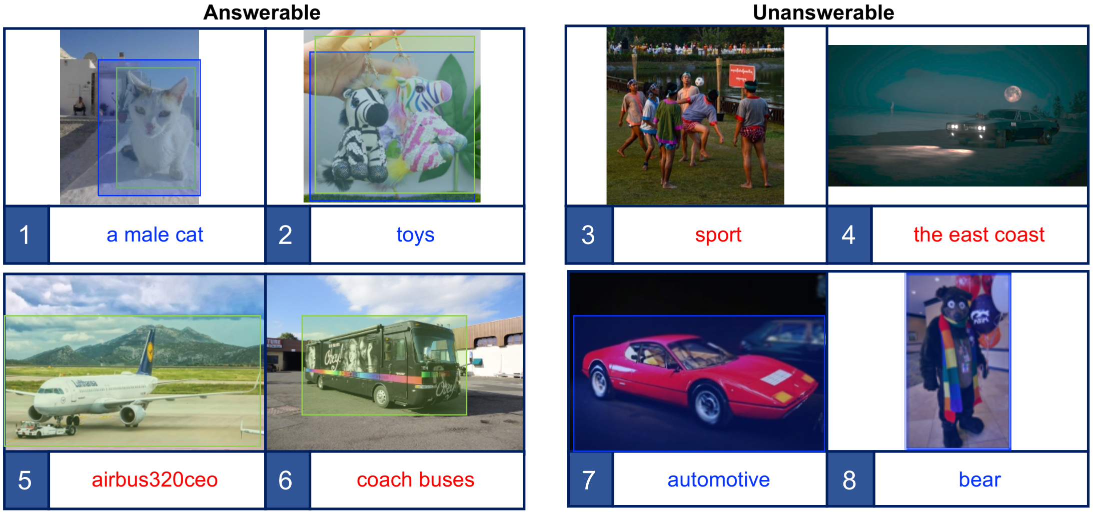


Figure 9: Examples of successful (top) and unsuccessful (bottom) visual grounding for answerable and unanswerable queries in the SMD4FVG dataset. The green and blue bounding boxes are ground-truth and the prediction of our model with the SNS setting, respectively.