Object Grounding via Iterative Context Reasoning

Lei Chen[1,2] , Mengyao Zhai[1,2] , Jiawei He[1,2] , Greg Mori[1,2]
[1]Simon Fraser University [2]Borealis AI
{chenleic, mzhai, jha203}@sfu.ca mori@cs.sfu.ca

Abstract

In this paper, we tackle the problem of weakly-supervised object grounding. For an image and a set of queries extracted from its description, the goal is to localize each query in the image. In a weakly-supervised setting, ground-truth query groundings are not accessible at training time. We propose a novel approach for weakly-supervised object grounding through iterative context reasoning in which we update query representations and region representations iteratively conditioning on each other. Such iterative contextual refinement gradually resolves ambiguity and vagueness in the queries and regions, thus helping to resolve challenges in grounding. We show the effectiveness of our proposed model on two challenging video object grounding datasets.

1. Introduction

Modern artificial intelligence systems focus heavily on extracting knowledge from visual and textual information captured from the real world. Promising progress has been made within the computer vision and natural language processing research communities. Such progress naturally leads to the emergence of various interdisciplinary tasks to bridge these two well-developed fields, such as image/video captioning [1, 25, 28], visual question answering [8, 24, 34], visual grounding [9, 20, 33], and text-based image/video generation [14, 18, 31].

In this paper we focus on the visual grounding task. Different from other tasks which usually relate visual and textual information on a holistic scale (e.g., one sentence describing the whole image/video), visual grounding aims to identify and localize objects mentioned in the textual description of the visual data. If the ground truth for grounding (regions in the image/video) is present at training time, learning the mapping between the region and text can be done in a straight-forward supervised manner. However, it is usually difficult to collect a sufficient amount of annotated data for a large set of queries, as required by deep learning based models. Therefore, a weakly supervised grounding approach that can be trained with only visual content and textual queries without the need for grounding annotation is desirable. Building such a model, if successful, can not only reduce the requirement for costly human labeling, but also take advantage of the abundant visual information aside from the grounding regions.

The core of visual grounding is to measure the similarity between textual queries and visual regions. Both queries and visual regions need to be embedded into a representation before a measurement can be applied across the two modalities. However, the levels of abstraction are usually different between elements from the two domains, making the similarity calculation difficult. An image region is typically detailed and specific while a textual query can be vague and ambiguous. An example is given in Fig. 1. When the goal is to ground the query *carrot* in a video frame, the query *carrot* may refer to a raw carrot, diced carrot or carrot soup. Existing approaches usually assign a single fixed representation to each query. The query encoded in this way would either focus on a single dominating meaning or end up as an average, mixing semantic meanings. Such query representations are not able to adapt well to varied visual instances with different appearance, in Fig. 1(a), when the query *carrot* focuses only on the raw carrot, it is more likely that the hand of the instructor is selected as its match.

Including context could be a solution to this problem. Since a query is eventually grounded to a region in the image, the semantics of the query should depend on the context of regions. As shown in Fig. 1(b), the query *carrot* should adapt to the visual content in the image and take a suitable representation for matching the diced carrot. In this paper, we propose to jointly encode the query and regions in an iterative manner so that the query and region representations could benefit from and adapt to each other for a more case-specific grounding task. Our model first embeds both query and region to an initial semantic space, and then iteratively refines the query representation based on the region context and refines the region representation based on the query context. With such iterative contextual representation refinement, the region and query are encoded with explicit awareness of each other, which gradually helps to resolve

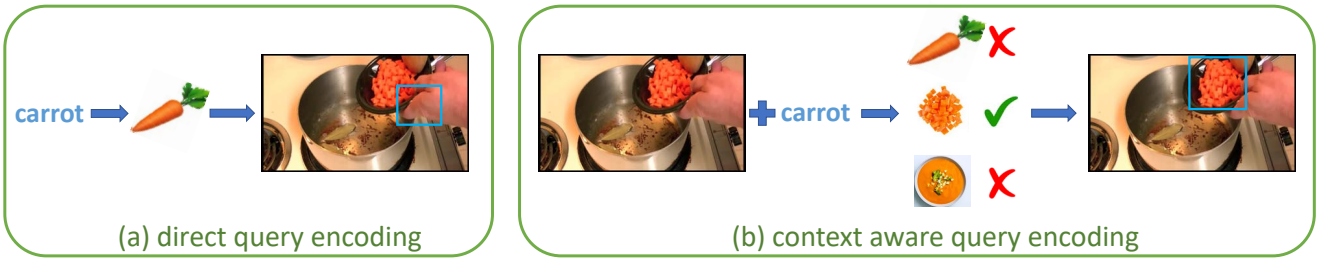

Figure 1. Mapping the query and image to the same semantic space for matching could be hard due to the ambiguity and vagueness of the query. A standalone encoding of the query might not adapt to all scenarios while conditioning the query on the visual content helps resolve ambiguity.

ambiguity and refine the representation for better grounding in each specific scenario.

2. Related Work

Bridging Visual and Textual Content. Natural language and computer vision are two fields that benefit greatly from deep convolutional neural networks and there are numerous works that try to bridge these two sources of information. Donahue et al. [5] contribute an early effort to exploit long-short term memory networks for generating textual descriptions for visual contents. Frome et al. [6] treat class labels in a classification task as an embedding in the semantic space via word2vec [15] and learn to embed the image close to its label in the semantic space, enabling classification for unseen class labels. Agrawal et al. [3] introduce the task of visual question answering, asking a neural network to comprehend a textual question and answer the question based on image content. Gao et al. [16] apply transformers to refine representations across textual and visual modalities for visual question answersing. Reed et al. [18] propose a generative network that takes in a sentence description and generates an image accordingly.

Visual Grounding. Visual grounding is a task that tries to establish precise correspondence between textual segments and image regions. Karpathy et al. [10] propose to use a multiple instance learning criterion and ranking loss for localizing objects in a sentence bounded by a dependency tree. In Karpathy et al. [9] the previous ranking loss is simplified and a bidirectional RNN is used to encode words in a sentence. Plummer et al. [17] propose the Flickr30k Entities dataset, providing annotations for the location of the noun phrases parsed from the corresponding captions of the Flickr30k dataset, and propose a Canonical Correlation Analysis based model for supervised phrase grounding. Rohrbach et al. [20] introduce the GroundeR model which uses an attention module to attend to proposals for each query and uses the attended region to reconstruct the query. The reconstruction loss requires no ground-truth labels and enables the model to work in both weakly-supervised and supervised settings. Xiao et al. [27] introduce a structural loss that exploits the parse tree of the image description to ground linguistic phrases in the form of spatial attention masks in a weakly-supervised setting. Zhou et al. [33] propose to use different weights for different frames when learning to ground objects in video frames, where the weight is decided by the frame grounding score and the query set for the frame. They also introduce an object grounding dataset of videos based on the YouCook2 dataset where the most common objects in the instructions are annotated with bounding boxes. Zhou et al. [32] propose to integrate grounding in generating video descriptions for better explainability and introduce the Activity Net Entities dataset which provides sparse object grounding annotations on top of the Activity Net Captions dataset. Hendricks et al. [2] integrate visual grounding techniques in learning a explanation generating agent so as to ground an agent's decision to certain visual attributes. These previous works would encode visual regions and textual queries independently and reason correspondence between elements from the two domains based on their independent features. Different from this pipeline, our model encodes visual regions and textual queries in a way such that they are aware of each other, resulting in refined visual and textual representations adapted to each specific scenario.

Graph Neural Networks. A vanilla feed forward neural network treats the inputs to the network equally without considering the structure among the inputs. Various graph neural network structures have been studied to model structure among the inputs. Scarselli et al. [22] introduce the Graph Neural Network (GNN) that takes in a graph of nodes and encodes each node based on the graph. Li et al. [12] bring gated recurrent units to GNNs for updating node representations which removes the constraints of parameters in GNNs and extends it to structured output. Zaheer et al. [30] propose the Deep Set to preserve set structure by using a set of permutation invariant operations before pooling the representations of nodes. Kipf et al. [11] propose the Graph Convolutional Network (GCN) as a localized first-order approximation of spectral graph convolutions. Since

then GCNs have been exploited in various vision tasks due to their scalability: Chen et al. [4] use GCNs for visual reasoning, Wang et al. [26] use GCNs to aggregate object information for video classification, Garcia et al. [21] use GCNs to reason within data batches for few shot recognition, Yao et al. [29] use GCN to integrate spatial information and semantics across regions in the task of image captioning.

3. Method

Given an image or video frame with a verbal description, object grounding is the task of localizing each object query mentioned in the description as a region in the image or video frame. Object queries are a set of predefined words and we select the top detections from a general detector as the candidate regions. In this section, we introduce our grounding method with iterative contextual reasoning. An overview of the model structure is provided in Fig. 2. After an initial semantic embedding for both regions and queries, we start to refine their representations jointly. Query representations will be updated conditioned on the context of regions and region representations will be updated conditioned on the queries. Such joint refinements are conducted iteratively for gradually resolving the ambiguity in the initial representations. In the weakly-supervised setting, the ground truth mappings between regions and queries are not provided as supervision in the training phase.

Initial Semantic Embedding. For an image with N_q queries $Q = \{q_i\}_{i=1}^{N_q}$ and N_r regions $R = \{r_i\}_{i=1}^{N_r}$, the queries are represented in a one-hot manner and the regions and corresponding visual features based on a pretrained network. Both region features and one-hot query encodings will be sent to a neural network to be embedded into an initial semantic space. We denote initial representations as $Q^{(0)} = [q_1^{(0)}, q_2^{(0)}, \ldots, q_{N_q}^{(0)}]$ and $R^{(0)} = [r_1^{(0)}, r_2^{(0)}, \ldots, r_{N_r}^{(0)}]$ respectively for queries and regions. The superscripts (0) are used to differentiate these representations from updated representations, to be discussed later.

Query Representation Refinement from Region Context. Conventional models conduct region-query matching directly with these independently obtained representations; the problem is that queries are usually ambiguous as they may refer to various objects of different appearances. Therefore, assigning a fixed representation for a query can not handle the ambiguity. To help resolve the ambiguity, the context of visual content in the image should play a role in shaping the representation of a query. This inspires us to update query representations conditioned on region representations to achieve a more precise query representation for each specific scenario. In addition, different queries may benefit from different visual information. To achieve that, each query should be granted the flexibility to pay attention to different regions so as to acquire different visual cues.

This conditional update of query representations can be viewed as sharing information between region representations and query representations. If we further take model representations as nodes in a graph, such passage of contextual information naturally forms a bipartite graph between the region representation nodes and the query representation nodes as illustrated in Fig. 3. Here we adopt the Graph Convolutional Network (GCN) [11] for updating the representation nodes via propagating information on this representation graph. The weight for each edge in the graph is determined based on the representations connected by the edge. Specifically, we denote the adjacency matrix in this query updating graph as $A_{r\to q}^{(0)}$, with the weight $A_{r\to q}^{(0)}(i,j)$ for the edge connecting $q_i^{(0)}$ and $r_j^{(0)}$ being simply their dot product $q_i^{(0)T} r_j^{(0)}$. The adjacency matrix has N_q rows and N_r columns as the information passes from region representation nodes to query representation nodes, the i^{th} row in the adjacency matrix consists of weights between $q_i^{(0)}$ and all region representations. Since these edge weights are not normalized and might lead to scale change in the propagation, we normalize the adjacency matrix $A_{r\to q}^{(0)}$ by applying softmax on each row:

$$\tilde{A}_{r\to q}^{(0)}[i,:] = \text{softmax}(A_{r\to q}^{(0)}[i,:]), \tag{1}$$

and finally $\tilde{A}_{r\to q}^{(0)}$ is the actual adjacency matrix used by the GCN. Once the adjacency matrix is obtained, the graph is determined and we update the query representation nodes as:

$$Q^{(1)} = \tilde{A}_{r\to q}^{(0)} R^{(0)} W_{gcn,q}^{(0)}, \tag{2}$$

where $Q^{(1)}$ denotes the updated query representations and $W_{gcn,q}^{(0)}$ denotes learnable weights in this query updating GCN.

The problem with directly updating representations with such a graph is that it abandons previous query representations completely. This is not desirable since the query representations should not be completely determined by the visual context. To alleviate such forgetting, a gated update is performed instead to preserve a portion of information from previous query representations in the update process. Here we denote the visual context obtained from the GCN as $H_q^{(0)}$, the actual updating process is formulated as:

$$H_q^{(0)} = \tilde{A}_{r\to q}^{(0)} R^{(0)} W_{gcn,q}^{(0)}, \tag{3}$$

$$z_q^{(0)} = \text{sigmoid}(W_{gate,q}^{(0)}[Q^{(0)}; H_q^{(0)}] + b_{gate,q}^{(0)}), \tag{4}$$

$$Q^{(1)} = z_q^{(0)} \odot Q^{(0)} + (1 - z_q^{(0)}) \odot H_q^{(0)}, \tag{5}$$

where $W_{gate,q}^{(0)}$ and $b_{gate,q}^{(0)}$ are weights for determining the gate $z_q^{(0)}$, and a sigmoid function is used to constrain $z_q^{(0)}$

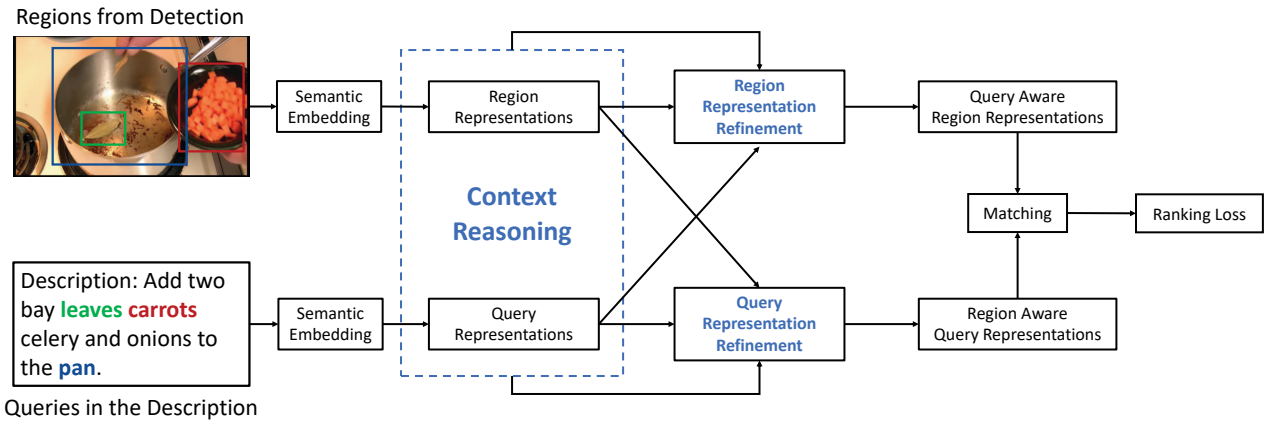

Figure 2. **Model Structure Overview**. Regions and their features are obtained from an external detector. Region features and one-hot encoded queries are first embedded to a semantic space as their initial representations. A contextual reasoning is performed between regions and queries. Query representations are updated with region context and region representations are updated with query context. This refinement is iterated for gradually resolving the ambiguity in representations for the two domains. At the end of the iterative refinement the correspondence between regions and query sets are computed upon their final representations and sent to a ranking loss.

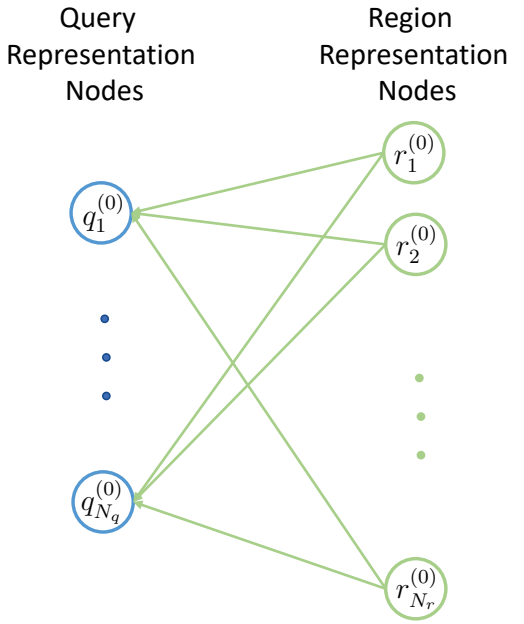

Figure 3. An illustration of the graph for updating query representations from region representations. Current region representations $r_1^{(0)}, r_2^{(0)}, \ldots, r_{N_r}^{(0)}$ and query representations $q_1^{(0)}, q_2^{(0)}, \ldots, q_{N_q}^{(0)}$ for an image and its query set are taken as nodes in the graph. The query representations are updated on condition of region representations which forms a bipartite graph between the two set of nodes. The weight on each edge is determined by the nodes it connects, allowing propagation of different region contexts when updating different query representations.

in $(0, 1)$. This gate will decide how much information to preserve from previous representations during this update.

Region Representation Refinement from Query Context. Similarly, the representations of regions can also benefit from the context of queries. Thus a similar graph-based update process is adopted to update the region representations with query context. Denote the raw adjacency matrix in this graph as $A_{q\to r}^{(0)}$, where the weight for each edge is also determined as the dot product of the two representations it connects. Note that different from the query-updating graph, this adjacency matrix has N_r rows and N_q columns, where the i^{th} row contains weights between $r_i^{(0)}$ and all query representations. And again it is normalized by applying softmax on each row:

$$\tilde{A}_{q\to r}^{(0)}[i,:] = \text{softmax}(A_{q\to r}^{(0)}[i,:]), \tag{6}$$

to obtain the normalized adjacency matrix $\tilde{A}_{q\to r}^{(0)}$. Denote the learnable weights in the GCN for updating region representations as $W_{gcn,r}^{(0)}$, the contextual information propagated from query representations as $H_r^{(0)}$, the learnable weights and bias for determining the gate $z_r^{(0)}$ as $W_{gate,r}^{(0)}$ and $b_{gate,r}^{(0)}$, and the updated region representations as $R^{(1)}$, the corresponding updating process for the region representations is formulated as:

$$H_r^{(0)} = \tilde{A}_{q\to r}^{(0)} Q^{(0)} W_{gcn,r}^{(0)}, \tag{7}$$

$$z_r^{(0)} = \text{sigmoid}(W_{gate,r}^{(0)}[R^{(0)}; H_r^{(0)}] + b_{gate,r}^{(0)}), \tag{8}$$

$$R^{(1)} = z_r^{(0)} \odot R^{(0)} + (1 - z_r^{(0)}) \odot H_r^{(0)}. \tag{9}$$

With the aforementioned two cross updates, the updated query representations and region representations are conditioned on each other as context. The possible explanation of a query representation is more likely to be restricted within

relevant visual content and a region representation is more likely to demonstrate the visual concepts that relate to the queries in the description.

Iterative Contextual Reasoning. The above process concludes one iteration of representation refinement. A new iteration of context reasoning and representation refinement could then start with the updated representations $Q^{(1)}$ and $R^{(1)}$. These new representations will become nodes in the new graph and the edges will be determined accordingly. Another two gated GCNs will be applied for another round of query and region representation refinement. Such an updating process could be repeated for continuing to refine the query and region representations so as to gradually resolve the ambiguity in the grounding task.

In each iteration, we focus on information propagation from regions to queries or vice versa; information propagation within regions or queries is not explicitly considered. However, when the query representation updated in context of regions participates in the next iteration of region representation refinement, the information starts to propagate within the regions, and similarly information will also start to propagate within the queries when the updated region representation takes part in the next query update.

Inference. After L iterations of contextual reasoning, the system would output the final representations $Q^{(L)} = [q_1^{(L)}, q_2^{(L)}, \ldots, q_{N_q}^{(L)}]$ and $R^{(L)} = [r_1^{(L)}, r_2^{(L)}, \ldots, r_{N_r}^{(L)}]$ for queries and regions in the image respectively. We measure the correspondence between a query q_i and region r_j with their final representation after contextual reasoning, the matching score $s(q_i, r_j)$ is defined as their dot product $q_i^{(L)T} r_j^{(L)}$. At inference time, for each query from an image, we select the region that matches best with the query as its grounding prediction.

Training. Assume we have a dataset $\mathcal{D}$ consisting of K images and corresponding queries $\{(Q_i, R_i)\}_{i=1}^{K}$. We adopt a similar ranking loss as proposed in [9] for training a grounding network in a weakly-supervised setting. We consider the matching score between an image and the set of queries as the average matching score over all the queries and their matches

$$S(Q, R) = \frac{1}{N_q} \sum_{i=1}^{N_q} \max_j s(q_i, r_j), \tag{10}$$

where N_q is the number of queries for the image. The ranking loss requires that paired images and query sets score higher than unpaired ones:

$$\mathcal{L}_{rank} = \frac{1}{K} \sum_{(Q,R)\in\mathcal{D}} (\max(0, S(\tilde{Q}, R) - S(Q, R) + \Delta) + \max(0, S(Q, \tilde{R}) - S(Q, R) + \Delta)), \tag{11}$$

where $\tilde{Q}$ and $\tilde{R}$ denote queries and regions from another image and Δ is the margin.

4. Experiments

4.1. Experimental Setup

Datasets. We evaluate our framework on two video datasets that provide object queries and grounding annotations, YouCook2-BoundingBox [33] and ActivityNet-Entities [32]. The YouCook2-BoundingBox dataset contains 2000 unconstrained cooking videos and a description is provided for segments in the videos. Bounding box annotations are provided for the 63 most frequent objects in the descriptions along with four referring expressions: *it, them, that, they*. ActivityNet-Entities contains 14281 videos which are also temporally annotated into captioned segments. Sparse grounding annotations are provided for 432 frequently appearing objects in segment captions.

To train our model, we sample 5 frames from each segment in YouCook2-BoundingBox and 10 frames from each segment in ActivityNet-Entities for a similar interval between frames. For Youcook2-BoundingBox dataset we follow the split of the dataset as in [33]. For ActivityNet-Entities dataset, the test data is hosted on an evaluation server which is not ready at the moment, so we split its validation sets randomly and evenly into two halves as our own validation and testing set. For the YouCook2-BoundingBox dataset we use 42900 frames for training, 60682 frames for validation and 28304 frames for testing, with an average of 1.2 visible and annotated queries per frame in the validation set. For the our split of the ActivityNet-Entities dataset there are 309000 frames for training, 37700 frames for validation and 39360 frames for testing, with an average of 0.27 visible and annotated queries per frame (due to the sparse annotation).

Comparing Approaches. We compare our method with two state-of-the-art grounding approaches, namely *GroundeR* [20] and *DVSA* [9]. For YouCook2-BoundingBox dataset we also include comparison with more recent results conducted on the dataset from Zhou et al. [33] and Shi et al. [23]. All methods for each dataset except [23] share the same region proposals and features for fair comparison.

Evaluation Metric. The evaluation metric is the bounding box localization accuracy as in [33, 32], the ratio of correctly grounded bounding boxes over all grounded boxes. A predicted bounding box is considered positive if it has an Intersection over Union (IOU) of larger than 0.5 with the ground-truth bounding box. We provide both the bounding box accuracies that are computed globally without distinction of different query classes and those averaged over the query classes. Queries that are not annotated or not visible are not considered when computing grounding accu-

Method	Box Accuracy		Average Box Accuracy	
	val	test	val	test
GroundeR[20]	-	-	19.63	19.94
DVSA[9]	-	-	30.51	30.80
Zhou et al. [33]	-	-	30.31	31.73
Shi et al. [23][1]	46.41	46.33	39.54	40.71
Contextual Reasoning(1 iter)	38.89	-	32.26	33.41
Contextual Reasoning(2 iter)	40.76	-	33.24	34.90

Table 1. Box Accuracy and Average Box Accuracy over query classes on YouCook2-BoundingBox dataset for different models. For our model, we demonstrate the performance for different iterations of contextual reasoning. We only report the box accuracy on validation set for our methods since the evaluation server could only evaluate the average box accuracy.

Method	Box Accuracy		Average Box Accuracy	
	val	test	val	test
GroundeR[20]	16.45	16.26	13.44	11.36
DVSA[9]	34.72	34.63	22.75	22.46
Contextual Reasoning(1 iter)	35.37	36.98	23.56	24.32
Contextual Reasoning(2 iter)	38.54	40.08	23.09	24.58

Table 2. Box Accuracy and Average Box Accuracy over query classes on ActivityNet-Entities dataset for different models. For our model, we demonstrate the performance for different iterations of contextual reasoning.

racy. For YouCook2-BoundingBox dataset the bounding boxes for the queries are annotated for frames sampled at 1fps in the validation and test set, while for ActivityNet-Entities the bounding box for each query is annotated only in one of the evenly sampled 10 frames in the corresponding video segment. The evaluation is conducted accordingly on these frames.

Implementation Details. For YouCook2-BoundingBox dataset we use the region proposals and features provided by [33] generated by a Faster-RCNN [19], the features are the 2048-dimensional output after the ROI pooling layer. For ActivityNet-Entities dataset we generate candidate regions from another Faster-RCNN [19] model trained for the MSCOCO detection task which has a Resnet-101 [7] and FPN [13] backbone, and use the detector's fc7 features as visual features.

For negative examples to be used in the ranking loss we randomly sample frames together with their queries from another video. The initial semantic embedding module has one embedding layer and one fully connected layer for embedding the queries, and one fully connected layer for embedding visual features, embedding both features to a 128-dimensional space. The contextual reasoning module is built after the initial embedding module by stacking gated-GCN networks, all the GCN networks keep the feature dimension at 128. The margin for the ranking loss is set to 0.6. An Adam optimizer with a learning rate of 0.0001 and weight decay of 0.0005 is used to optimize all the models. For each model, we pick the epoch where it gives lowest loss on our validation set and report the grounding accuracy on both validation and test set.

4.2. Object Grounding on YouCook2-BoundingBox

The grounding accuracies of different models are shown in Table 1. As Zhou et al. [33] use only average box accuracy as a metric, the global box accuracies are not available for their model and the two baselines. Since the evaluation server also only evaluates average box accuracy, we could only report box accuracy for our models on the validation set. The *DVSA* model gives an average box accuracy of 30.80% on the test set. With one iteration of contextual update of feature and query representations our model improved this performance by 2.61% on the test set and outperforms [33]. With an extra iteration of representation update the performance is further increased by 1.49%. The performance boost for global box accuracy is larger than that for average box accuracy, indicating that queries that appear more frequently in the dataset benefit more from context reasoning. This is reasonable since the more frequently mentioned query classes are more likely to appear in varied forms, thus benefitting more from the resolution of ambiguity. Adding more stacks of contextual updates of representations does not give extra boost to the performance. The Grounding by Attention model gives

[1]Shi et al. [23] use different region proposals than other methods here. Although also built on DVSA like Zhou et al. [33] and our approach, they report much higher DVSA performance, with global box accuracy of 44.26% and 44.16% for validation and test set and average box accuracy of 36.90% and 37.55% for validation and test set.

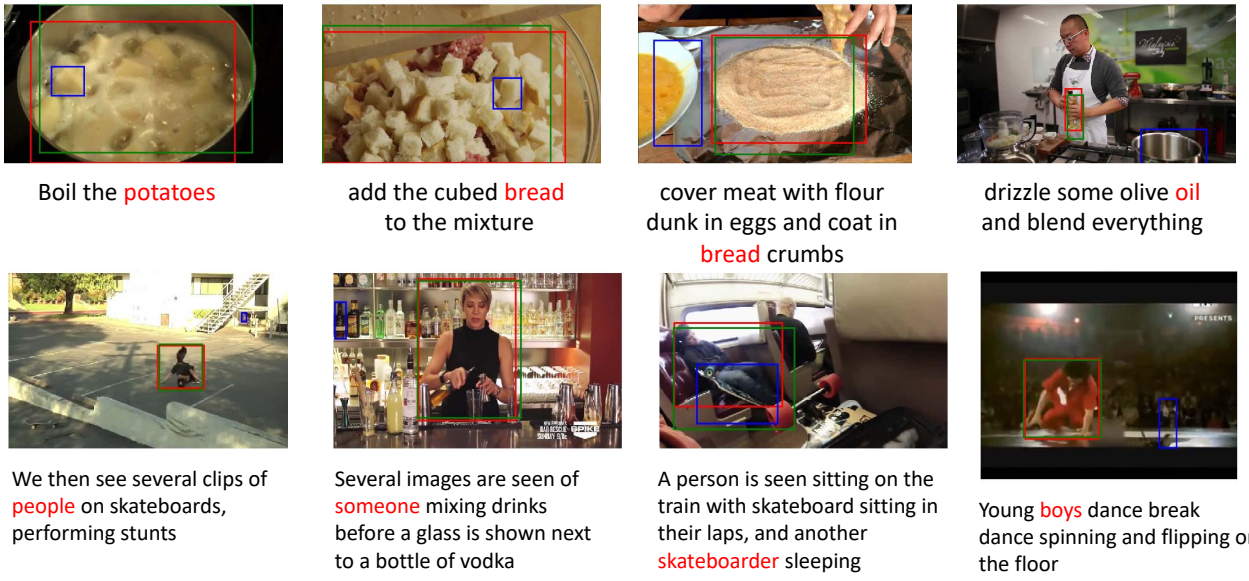

Figure 4. Qualitative results on YouCook2-BoundingBox (first row) and ActivityNet-Entities (second row). In each figure, the ground truth is denoted by a red bounding box, the prediction from base grounding by ranking model in blue and prediction from our model in green.

an accuracy of 19.94% which is behind the ranking based grounding models. This might result from the fact that the queries here are only object labels rather than noun phrases as in [20] thus the query reconstruction loss is less powerful, while our model tackles particularly the problem of vague queries by conditioning representations on context.

Qualitative comparisons are given in Fig. 4. Our model is able to ground the queries to the objects in various forms, while the base grounding by ranking model fails to recognize/localize objects in less typical appearance. We also inspect the queries and find that our final model gives largest performance boost to the query *salad*, *it* and *milk*. The query *salad* is a good example of a vague query as salad takes various appearances based on its ingredients, let alone the query *it* which could refer to anything in the frame. The query *milk* benefits from our model because it takes different appearance in different containers.

4.3. Object Grounding on ActivityNet-Entities

The results on the ActivityNet-Entities dataset present a similar trend as those in the previous experiment. Since the query set is substantially larger than that of YouCook2-BoundingBox and the scenes are more complicated, the box localization accuracy of all methods are lower. The *DVSA* model gives an average box accuracy of 22.46% on the test set. One iteration of our contextual update of the query and region representations improves the grounding accuracy by 1.94% while a second iteration further boosts the performance marginally by 0.26%. A look into the global box accuracy demonstrates that the second iteration of context reasoning still boosts performance on this metric. The difference of the performance boost for average and global performance indicates that the ActivityNet-Entities dataset may have more biased query distribution than YouCook2-BoundingBox and the model is helping more on the more dominating query classes. The *GroundeR* baseline is still left behind by ranking based models. In addition to the vague query problem mentioned above, sometimes in the ActivityNet-Entities dataset multiple queries correspond to the same object, this might confuse the *GroundeR* model by forcing the same region to reconstruct different queries.

Some qualitative results can be found in Fig. 4, again our model successfully localizes these queries despite the unusual appearance of the objects while the base model fails to adapt the queries to these scenes. As we inspected the performance for each category, we found that "skateboarder", "parking", "runner", "herself" and "background" are the categories whose performance enjoy the greatest boost from contextual reasoning.

5. Conclusion

In this paper, we tackled the problem of weakly supervised object grounding, where only the object queries are provided at training time without access to grounding annotation. We propose a weakly supervised grounding model that automatically represents a query and a region in context of each other with iterative refinement, which resolves the ambiguity between an abstract query and a specific region. We apply our grounding approach on the YouCook2-BoundingBox and ActivityNet Entities datasets, on which our model outperforms state of the art grounding by rank-

ing and grounding by attention models and can iteratively boost its performance via context reasoning.

References

[1] P. Anderson, X. He, C. Buehler, D. Teney, M. Johnson, S. Gould, and L. Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 1

[2] L. Anne Hendricks, R. Hu, T. Darrell, and Z. Akata. Grounding visual explanations. In *European Conference on Computer Vision (ECCV)*, 2018. 2

[3] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. Lawrence Zitnick, and D. Parikh. Vqa: Visual question answering. In *IEEE International Conference on Computer Vision (ICCV)*, 2015. 2

[4] X. Chen, L.-J. Li, L. Fei-Fei, and A. Gupta. Iterative visual reasoning beyond convolutions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 3

[5] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 2

[6] A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, T. Mikolov, et al. Devise: A deep visual-semantic embedding model. In *Advances in Neural Information Processing Systems (NIPS)*, 2013. 2

[7] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 6

[8] J. Johnson, B. Hariharan, L. v. d. Maaten, L. Fei-Fei, C. L. Zitnick, and R. Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1

[9] A. Karpathy and L. Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 1, 2, 5, 6

[10] A. Karpathy, A. Joulin, and L. F. Fei-Fei. Deep fragment embeddings for bidirectional image sentence mapping. In *Advances in Neural Information Processing Systems (NIPS)*, 2014. 2

[11] T. N. Kipf and M. Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*, 2017. 2, 3

[12] Y. Li, D. Tarlow, M. Brockschmidt, and R. Zemel. Gated graph sequence neural networks. In *International Conference on Learning Representations (ICLR)*, 2015. 2

[13] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie. Feature pyramid networks for object detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 6

[14] E. Mansimov, E. Parisotto, J. Ba, and R. Salakhutdinov. Generating images from captions with attention. In *International Conference on Learning Representations (ICLR)*, 2016. 1

[15] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems (NIPS)*, 2013. 2

[16] G. Peng, H. Li, H. You, Z. Jiang, P. Lu, S. Hoi, and X. Wang. Dynamic fusion with intra-and inter-modality attention flow for visual question answering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

[17] B. A. Plummer, L. Wang, C. M. Cervantes, J. C. Caicedo, J. Hockenmaier, and S. Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *IEEE International Conference on Computer Vision (ICCV)*, 2015. 2

[18] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee. Generative adversarial text to image synthesis. In *International Conference on Machine Learning (ICML)*, 2016. 1, 2

[19] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2015. 6

[20] A. Rohrbach, M. Rohrbach, R. Hu, T. Darrell, and B. Schiele. Grounding of textual phrases in images by reconstruction. In *European Conference on Computer Vision (ECCV)*, 2016. 1, 2, 5, 6, 7

[21] V. G. Satorras and J. B. Estrach. Few-shot learning with graph neural networks. In *International Conference on Learning Representations (ICLR)*, 2018. 3

[22] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini. The graph neural network model. *IEEE Transactions on Neural Networks*, 2009. 2

[23] J. Shi, J. Xu, B. Gong, and C. Xu. Not all frames are equal: Weakly-supervised video grounding with contextual similarity and visual clustering losses. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 5, 6

[24] M. Tapaswi, Y. Zhu, R. Stiefelhagen, A. Torralba, R. Urtasun, and S. Fidler. Movieqa: Understanding stories in movies through question-answering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 1

[25] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: A neural image caption generator. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 1

[26] X. Wang and A. Gupta. Videos as space-time region graphs. In *European Conference on Computer Vision (ECCV)*, 2018. 3

[27] F. Xiao, L. Sigal, and Y. Jae Lee. Weakly-supervised visual grounding of phrases with linguistic structures. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2

[28] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *International Conference on Machine Learning (ICML)*, 2015. 1

[29] T. Yao, Y. Pan, Y. Li, and T. Mei. Exploring visual relationship for image captioning. In *European Conference on Computer Vision (ECCV)*, 2018. 3

[30] M. Zaheer, S. Kottur, S. Ravanbakhsh, B. Poczos, R. R. Salakhutdinov, and A. J. Smola. Deep sets. In *Advances in Neural Information Processing Systems (NIPS)*, 2017. 2

[31] H. Zhang, T. Xu, and H. Li. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *IEEE International Conference on Computer Vision (ICCV)*, 2017. 1

[32] L. Zhou, Y. Kalantidis, X. Chen, J. J. Corso, and M. Rohrbach. Grounded video description. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2, 5

[33] L. Zhou, N. Louis, and J. J. Corso. Weakly-supervised video object grounding from text by loss weighting and object interaction. In *British Machine Vision Conference (BMVC)*, 2018. 1, 2, 5, 6

[34] Y. Zhu, O. Groth, M. Bernstein, and L. Fei-Fei. Visual7w: Grounded question answering in images. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 1