Shifting More Attention to Visual Backbone: Query-modulated Refinement Networks for End-to-End Visual Grounding

Jiabo Ye[1] Junfeng Tian[2] Ming Yan[2] Xiaoshan Yang[3]
Xuwu Wang[4] Ji Zhang[2] Liang He[1] Xin Lin[1]
[1]East China Normal University, Shanghai, China [2]Alibaba Group, Hangzhou, China
[3]NLPR, CASIA, Beijing, China [4]Fudan University, Shanghai, China
jiabo.ye@stu.ecnu.edu.cn, {xlin,lhe}@cs.ecnu.edu.cn, xwwang18@fudan.edu.cn
xiaoshan.yang@nlpr.ia.ac.cn, {tjf141457,ym119608,zj122146}@alibaba-inc.com

A. Dataset Analysis

We present the statistics about five datasets we used in our experiments. Table 1 shows the statistics.

ReferItGame: ReferItGame [5] contains images from SAIAPR12 [2] and collects expressions by a two-player game. In this game, the first player is shown an image with an object annotation and asked to write a natural language expression referring to the object. The second player is shown the same image and the written expression and asked to click the object's corresponding area. If the clicking is correct, the two players receive points and swap roles. If not, the new image would be presented.

RefCOCO, RefCOCO+, RefCOCOg: These three datasets contain images from MSCOCO [6]. Expressions in RefCOCO [12] and RefCOCO+ [12] are also collected by the two-player game proposed in ReferitGame [5]. There are two test splits called "testA" and "testB". Images in "testA" only contain multiple people annotation. In contrast, images in "testB" contain all other objects. Expressions in RefCOCOg [7] are collected on Amazon Mechanical Turk in a non-interactive setting. Thus, the expressions in RefCOCOg are longer and more complex. RefCOCOg has "google" and "umd" splits. The "google" split does not have a public test set. And there is an overlap between the training and validation image set. The "umd" split does not have this overlap. We denote these splits by "val-g", "val-u" and "test-u", respectively.

Flickr30k Entities: Flickr30k Entities [9] contains images in Flickr30k dataset. The query sentences are short noun phrases in the captions of the image. The queries are easier to understand.

Dataset	Images	Instances	Queries
RefCOCO	19,994	50,000	142,210
RefCOCO+	19,992	49,856	141,564
RefCOCOg	25,799	49,822	95,010
ReferItGame	20,000	19,987	120,072
Flickr30K Entities	31,783	427,000	427,000

Table 1. Data statistics of RefCOCO, RefCOCO+, RefCOCOg, ReferItGame and Flickr30K Entities.

Model	ReferItGame	
	val	test
DarkNet53		
ReSC-Base	66.78	64.33
+QD-ATT	$67.85_{+1.07}$	$65.21_{+0.88}$
ResNet50		
TransVG	71.60	69.76
+QD-ATT	$75.01_{+3.41}$	$72.67_{+2.91}$

Table 2. Performance of ReSC and TransVG when QD-ATT is adopted.

B. QD-ATT in Other Vision Backbones

The experimental results in our paper have shown that the Query-aware Dynamic Attention can help improve the transformer-based visual backbone. In this section, we present the performance improvement in visual grounding by applying the Query-aware Dynamic Attention to DarkNet53 [10] and ResNet [3] to show its generalizability in different backbones.

DarkNet53. We use a state-of-the-art one-stage visual grounding framework ReSC [11], which adopts the visual feature map from the 102-th convolutional layer of DarkNet53. We append Query-aware Dynamic Attention modules after the 78-th, 91-th and 102-th convolutional layer of DarkNet53. We take the visual feature map from the last Query-aware Dynamic Attention module as the output of the visual backbone.

ResNet. We consider the TransVG [1] framework, which uses ResNet50 as the visual backbone. We apply Query-aware Dynamic Attention modules to the C2, C3, C4, and C5 layers.

The evaluation results using different visual backbones on the ReferItGame dataset are shown in Table 2. The proposed QD-ATT can also significantly improve the performance.

C. Ablation study on RefCOCO

We also present the ablation study results on RefCOCO dataset in Table 3 and Table 4. The experimental results reveal similar conclusions. We also notice that spatial attention is more useful in "testA" split and channel attention is more useful in "testB" split. It is because the referents in "testB" can be arbitrary objects but images in "testA" only contain multiple people annotation.

	RefCOCO	
	testA	testB
(a)	84.63	81.42
(b)	84.46	80.96
(c)	84.01	79.83

Table 3. Ablation studies of QD-ATT in two phases of QRNet.

	RefCOCO	
	testA	testB
(d)	84.61	81.25
(e)	85.43	80.91

Table 4. Ablation studies of different attentions in QD-ATT.

D. Analysis of hyper-parameter K

We study the impact of hyper-parameter K which denotes the factor dimension. We report the accuracy of our model with $K = \{10, 30, 50, 70\}$ in Figure 1. When K is small, the QD-ATT cannot learn the transformation well. And when K is large, the QD-ATT is easy to overfit and performs worse on validation set. We take the best $K = 30$ on the validation set as our default setting.

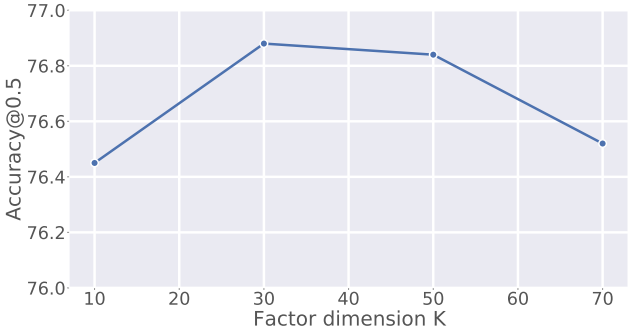

Figure 1. The effect of factor dimension K on the validation split of ReferItGame dataset.

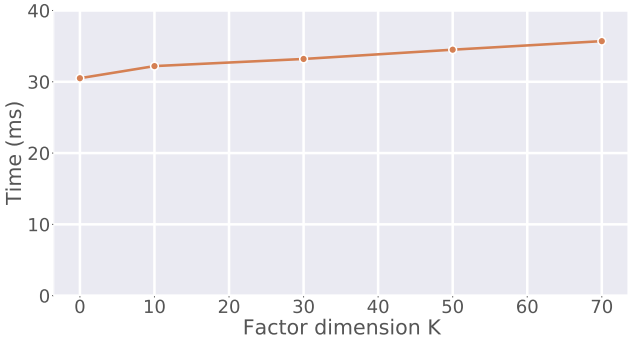

Figure 2. The running time (ms) of our model with different factor dimension K on the validation split of ReferItGame.

E. Efficiency Analysis

We study the running time of our model with different factor dimensions in the Query-aware Dynamic Attention. We set $K = \{0, 10, 30, 50, 70\}$. $K = 0$ is a special setting which means the Query-aware Dynamic Attention directly returns the input feature without any computation. We conduct the experiment on a single NVIDIA RTX3090 GPU. The results are shown in Figure 2. We notice that the running time increases linearly with the growth of K. We also observe that incorporating the Query-aware Dynamic Attention does not introduce excessive computation cost, which means our QD-ATT is efficient and effective.

F. Qualitative Analysis

Feature Refinement Visualization. We present more visualize cases in Figure 3. It shows the ability of our QRNet to refine general purposed feature maps to query-aware feature maps.

Comparison with TransVG. We quantitatively compare the results predicted by our model and the TransVG in Figure 4. Green boxes are the ground truth. Yellow and red boxes are the predictions of our model and TransVG, respectively. In the first row, we notice that the predictions of

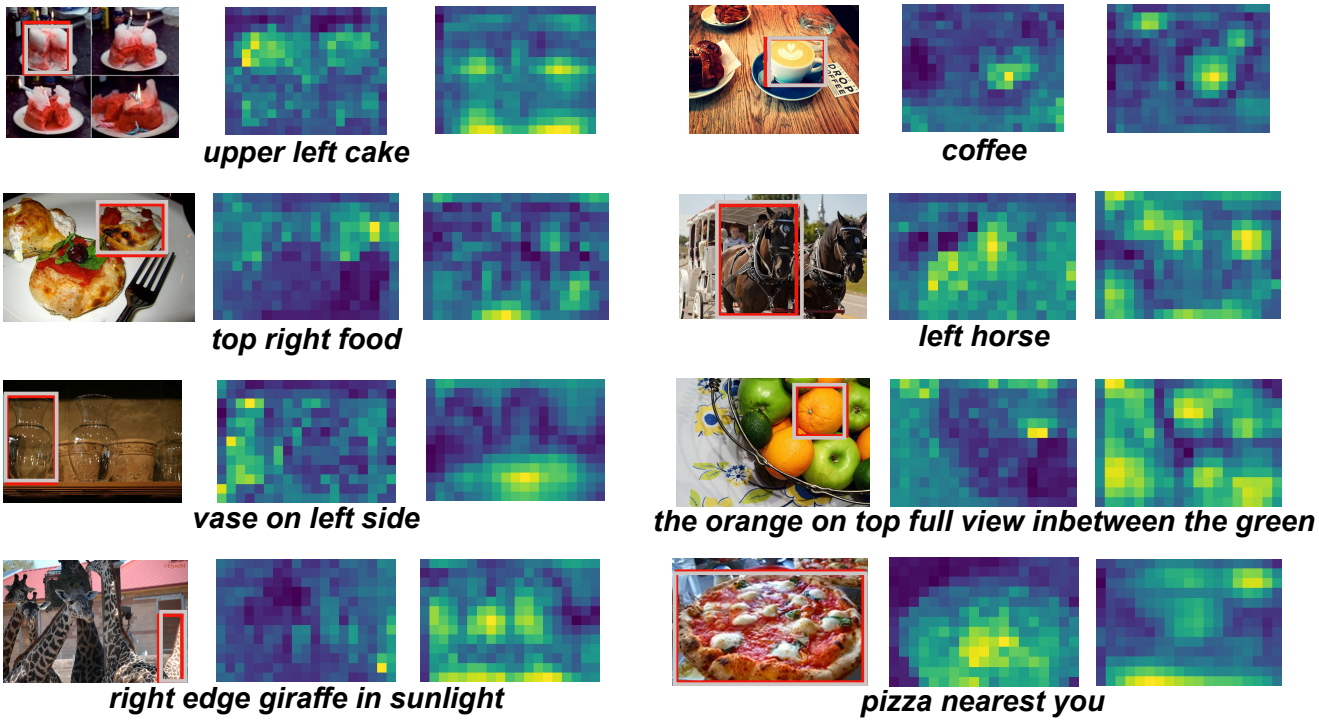

Figure 3. Visualization of activation maps from the backbone of our QRNet and original Swin-Transformer. Red: the predicted box. White: the ground-truth box.

TransVG are close to the ground truth. However, sometimes the TransVG only localizes the significant areas of the target objects (e.g., "mountain on right"), and sometimes the predicted boxes cover the irrelevant areas around the target objects (e.g., "person in white"), resulting in failures. In the second row, the TransVG does not fully capture the semantics of queries and is vulnerable to the interference of significantly noisy objects in the image, resulting in incorrect predictions. Our model utilizes query sentence representation to guide the visual backbone to obtain query-aware visual features, which helps the framework give correct predictions.

Failure Cases. We show some typical failure cases in Figure 5. One type of error is the long tail description. For the first example, our model does not recognize "slats" correctly. During both the pre-training the fine-tuning, the samples of long-tail concepts are infrequent. Thus, our method fails. For the second image, we notice that "crazy" is an abstract word, and its visual appearance can be completely different in different contexts. Therefore, it is also hard for our model to learn. In the third example, we found that our model understands the query's intention, but it tends to predict a box containing some objects, and the annotator does not have such bias. For the last example, we find that our model may make mistakes when facing long sentence queries. Our model only takes the [CLS] representation rather than the complete sequence representation from BERT to refine the visual features, limiting the ability to understand long text.

G. Limitation

In this paper, we propose QD-ATT to extract query-consistent features from the visual backbone. However, we only take the contextual textual representation of the query to keep it efficient. Due to the lack of fine-grained multimodal interaction, we still need a post-interaction module in the visual grounding framework. It also hinders generalizing our module to the task requiring complex reasoning, *e.g.*, multimodal dialogue [8] and visual storytelling [4].

H. Online Deployment Setting

We apply QRNet to enhance the search engine in Pailitao at Alibaba and perform A/B testing. In order to reduce the discrepancy between the images from consumers and sellers, it is important to locate the target from the item image. Note that a detection-based two-stage method is already deployed online. The A/B testing system split online users equally into two groups and direct them into two separate buckets, respectively. Then in bucket A (i.e., the baseline group), the target boxes from images are generated by a

Figure 4. Qualitative comparison with TransVG. Green boxes are the ground-truth, yellow boxes are the predictions of our model, and red boxes denote the predictions of TransVG.

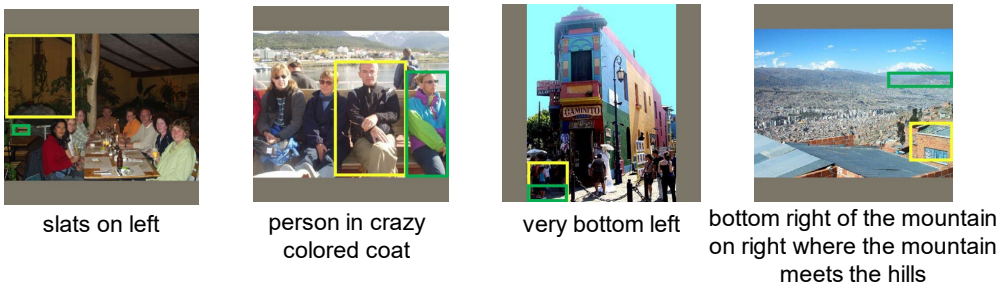

Figure 5. The failure cases of our model. Green boxes are the ground-truth, yellow boxes are the predictions of our model.

detection-based method. While for users in the B bucket (i.e., the experimental group), the target boxes are generated by the proposed QRNet. All the conditions of the two buckets are identical except for the target grounding methods.

We calculate the clicked query rate and the number of transactions for all the categories in the two buckets. The online A/B testing lasts for two days, and over 1.5 million daily active users are in each bucket. We find that the performance in the experimental bucket (i.e., QRNet) is significantly better ($p < 0.05$) than that in the baseline bucket (i.e., detection-based) on both measures. QRNet decreases the no click rate by 1.47% and improves the number of transactions by 2.20% over the baseline. More specifically, the decrease of the no click rate implies that QRNet can generate more accurate target boxes so that users are more likely to click. The improvement of the number of transactions means that the clicked item is exactly what the users want to purchase, which also reveals the great performance of QRNet.

I. Pseudo-code of QD-ATT

We present a Pytorch-style pseudo-code in Listing 1.

References

[1] Jiajun Deng, Zhengyuan Yang, Tianlang Chen, Wengang Zhou, and Houqiang Li. Transvg: End-to-end visual grounding with transformers. *arXiv preprint arXiv:2104.08541*, 2021. 2

[2] Hugo Jair Escalante, Carlos A Hernández, Jesus A Gonzalez, Aurelio López-López, Manuel Montes, Eduardo F Morales, L Enrique Sucar, Luis Villasenor, and Michael Grubinger. The segmented and annotated iapr tc-12 benchmark. *Computer vision and image understanding*, 114(4):419–428, 2010. 1

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1

[4] Ting-Hao Huang, Francis Ferraro, Nasrin Mostafazadeh, Ishan Misra, Aishwarya Agrawal, Jacob Devlin, Ross Girshick, Xiaodong He, Pushmeet Kohli, Dhruv Batra, et al. Visual storytelling. In *Proceedings of the 2016 Conference of*

```
# Perform linear transform to change the dimension of the vector space from in_dim to out_dim
# Inputs:
#   F: the input visual features (B, H, W, D_in)
#   q: the input linguistic feature (B, D_q)
# Return:
#   F_q: the output transformed visual features (B, H, W, D_out)
# Parameters:
#   linear: a plain linear layer to generate factor U
#   S: a random-initialized factor matrix (K, D_out)
def dynamic_linear(F, q):

  # Parameter generation
  # K is the  factor dimension

  # generate the factor matrix U
  U = reshape(linear(q), (B, D_in+1, K))
  # reconstructed matrix by U x S
  M = einsum("bik,kj->bij", U, S)
  Weight, bias = M[:,:-1], M[:,-1]

  # Perform linear transform
  F_q = einsum("b...d,bde->b...e", F, Weight) + bias
  return F_q

# Perform dynamic attention on visual feature map by textual guidance.
# Inputs:
#   F: input visual feature map (B, H, W, C)
#   q: [CLS] token's last hiddenn states from BERT (B, D_q)
# Return:
#   F_refined: refined visual feature (B, H, W, C)
def query_aware_dynamic_attention(F, q):
    # Channel Attention
    B,H,W,C = F.shape
    F_flatten = reshape(F, (B, -1, C))
    avg_pool = mean(F_flatten, dim=1)
    # The mlp transform the channel dimension  C -> C/16 -> C
    att_avg = dynamic_linear(relu((dynamic_linear(avg_pool, q))), q)
    max_pool = max(F_flatten, dim=1)
    att_max = dynamic_linear(relu((dynamic_linear(max_pool, q))), q)
    attn = sigmoid(att_avg + att_max)
    F = F * reshape(attn, (B, 1, 1, C))
    # Spatial Attention
    # dynamic_linear transform channel dimension C -> 1
    attn = sigmoid(dynamic_linear(F, q))
    F_refined = F * attn
    return F_refined
```

Listing 1. Pytorch-style pseudo-code of Query-aware Dynamic Attention

the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 1233–1239, 2016. 3

[5] Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. 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, 2014. 1

[6] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 1

[7] Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L Yuille, and Kevin Murphy. Generation and comprehension of unambiguous object descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 11–20, 2016. 1

[8] Liqiang Nie, Wenjie Wang, Richang Hong, Meng Wang, and Qi Tian. Multimodal dialog system: Generating responses via adaptive decoders. In *Proceedings of the 27th ACM International Conference on Multimedia*, pages 1098–1106, 2019. 3

[9] Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *Proceedings of the IEEE international conference on computer*

vision, pages 2641–2649, 2015. 1

[10] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018. 1

[11] Zhengyuan Yang, Tianlang Chen, Liwei Wang, and Jiebo Luo. Improving one-stage visual grounding by recursive sub-query construction. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16*, pages 387–404. Springer, 2020. 2

[12] Licheng Yu, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara L Berg. Modeling context in referring expressions. In *European Conference on Computer Vision*, pages 69–85. Springer, 2016. 1