The Thirty-Seventh AAAI Conference on Artificial Intelligence (AAAI-23)

Referring Expression Comprehension Using Language Adaptive Inference

Wei Su[1], Peihan Miao[2], Huanzhang Dou[1], Yongjian Fu[1], Xi Li[1,3,4]*

[1]College of Computer Science & Technology, Zhejiang University
[2]School of Software Technology, Zhejiang University
[3]Shanghai Institute for Advanced Study, Zhejiang University
[4]Shanghai AI Laboratory
{weisuzju, peihan.miao, hzdou, yjfu, xilizju}@zju.edu.cn

Abstract

Different from universal object detection, referring expression comprehension (REC) aims to locate specific objects referred to by natural language expressions. The expression provides high-level concepts of relevant visual and contextual patterns, which vary significantly with different expressions and account for only a few of those encoded in the REC model. This leads us to a question: *do we really need the entire network with a fixed structure for various referring expressions?* Ideally, given an expression, only expression-relevant components of the REC model are required. These components should be small in number as each expression only contains very few visual and contextual clues. This paper explores the adaptation between expressions and REC models for dynamic inference. Concretely, we propose a neat yet efficient framework named Language Adaptive Dynamic Subnets (LADS), which can extract language-adaptive subnets from the REC model conditioned on the referring expressions. By using the compact subnet, the inference can be more economical and efficient. Extensive experiments on RefCOCO, RefCOCO+, RefCOCOg, and Referit show that the proposed method achieves faster inference speed and higher accuracy against state-of-the-art approaches.

Introduction

Referring expression comprehension (REC) (Deng et al. 2021; Liao et al. 2020; Yang et al. 2020) aims at locating the target object in an image according to a natural language description, which can facilitate not only human-machine interaction in the physical world but also the advancement of tasks such as visual question answering (Li et al. 2018) and image retrieval (Salvador et al. 2016). Different from object detection (Carion et al. 2020), which only localizes objects of predefined categories, REC needs to utilize the characteristics of the target object described by referring expressions, including visual patterns (*e.g.* category and attribute) and contextual patterns (*e.g.* relative position and relationship), to locate the most matching region in the image.

As the referring expressions exclusively provide the visual and contextual clues about the referred objects, we overview the REC datasets and summarize two main characteristics of the expressions: (1) *the abundance of semantic information varies widely for different expressions*, and (2) *the visual and contextual patterns required for each expression are only a tiny fraction of the pattern set corresponding to the dataset*. Taking Fig. 1 as an example, the image is assigned three referring expressions, and the complexity of the expressions varies greatly. In terms of the first characteristic, the second expression in Fig. 1 contains abundant information about the referred object, including three categories (”girl”, ”father”, ”shirt”), two attributes (”small”, ”green”), and the relationship (”held”). Conversely, the third expression is less informative, containing only one category (”car”) and two attributes (”black”, ”back”). In terms of the second characteristic, there are only 1-3 categories in each expression in Fig. 1, compared to 80 categories in RefCOCO (Yu et al. 2016) and 238 categories in Referit (Kazemzadeh et al. 2014), which indicates that the category-relevant visual patterns required for each expression are only a tiny proportion of those corresponding to the entire dataset. Similarly, the contextual patterns also account for a very low proportion.

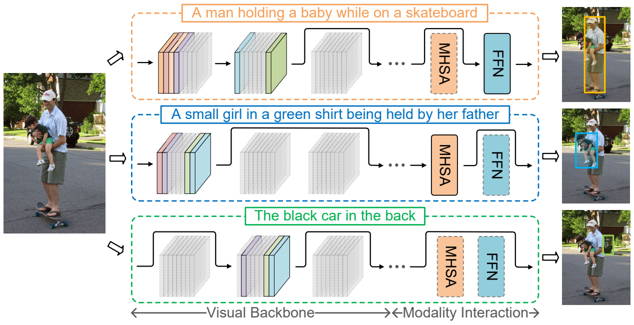
A man holding a baby while on a skateboard
MHSA
FFN
A small girl in a green shirt being held by her father
MHSA
FFN
The black car in the back
MHSA
FFN
Visual Backbone
Modality Interaction

Figure 1: The schematic of the proposed language adaptive dynamic subnet framework (LADS). Given an image and several referring expressions, the LADS framework can select relevant components (*i.e.* filters and layers) from the entire REC network to construct language-adaptive subnets.

The diversity and low informativeness of referring expressions lead us to an interesting research question: **Is the entire REC network with a fixed structure necessary for various referring expressions?** Intuitively, we need a dynamic inference framework in which we can construct a language-adaptive REC network on the fly, conditioned

*indicates corresponding author.

on different referring expressions. The dynamic networks should be compact and light-weighted since the referring expressions contain very few visual and contextual clues/-patterns. To this end, we utilize the entire REC network as a supernet, which can be viewed as an ensemble of subnets for various referring expressions. During inference, only the expression-relevant subnets are extracted from the supernet as the specific REC models, and the expression-irrelevant subnets are removed, which is illustrated in Fig. 1.

Concretely, we propose a neat yet efficient framework named Language Adaptive Dynamic Subnets (LADS), which can adaptively extract REC subnets from the supernet conditioned on different referring expressions. To obtain more flexible and compact subnets, we also propose to select layers and filters from the REC supernet jointly, and apply mutual information to constrain the alignment of subnets and referring expressions. Specifically, the linguistic feature of each referring expression is extracted first and then mapped to binary selection gates. The selection gates are used to pick out layers and filters from the REC supernet, and construct a compact subnet dedicated to the specific referring expression. It should be noted that the subnets maybe cohere with each other in the initial REC supernet. During training, LADS tries to strengthen the expression-relevant subnets and mitigate interference between subnets selected for different expressions. After training, the subnets can be extracted directly for inference without retraining, just like the once-for-all network (Cai et al. 2019). Extensive experiments on RefCOCO (Yu et al. 2016), RefCOCO+ (Yu et al. 2016), RefCOCOg (Mao et al. 2016), and Referit (Kazemzadeh et al. 2014) show that the proposed framework achieves faster inference speed and higher accuracy compared to state-of-the-art methods, which shows the effectiveness of the LADS framework and its prospects in more compact and real-time REC inference.

The main contributions are summarized as follows:

- To achieve efficient dynamic reasoning in REC, we propose the Language Adaptive Dynamic Subnets (LADS) framework, which can adaptively select compact REC subnets conditioned on the referring expressions.
- To obtain flexible and compact subnets, we propose to select layers and filters from the REC supernet jointly for more diverse subnets in depth and width, and apply mutual information to constrain the alignment of subnets and referring expressions for more compact subnets.
- Experiments on four representative datasets show that the proposed method achieves superior inference speed and accuracy compared to state-of-the-art methods, with the language-adaptive compact REC subnets.

Related Work

Referring Expression Comprehension

Most conventional REC methods consist of two stages (Yu et al. 2018; Chen et al. 2021; Hong et al. 2019; Liu et al. 2019). In the first stage, the candidate regions of the input image are obtained by pre-trained object detectors (Ren et al. 2015; He et al. 2017). In the second stage, given a referring expression, the best matching candidate region is chosen as the prediction. Most two-stage methods improve overall performance through the second stage. They exploit linguistic and visual features for contextual information mining, and additionally leverage attributes (Liu, Wang, and Yang 2017; Yu et al. 2018), object relationships (Wang et al. 2019; Yang, Li, and Yu 2019), phrase co-occurrence (Bajaj, Wang, and Sigal 2019), *etc.* to improve the performance. Despite the remarkable success, the overall performance of the two-stage methods is limited by the accuracy and speed of candidate region generation in the first stage. Recently, one-stage methods (Liao et al. 2020; Yang et al. 2020; Deng et al. 2021; Zhou et al. 2021; Luo et al. 2020) have proliferated and achieved excellent performance. In the one-stage paradigm, the network densely fuses linguistic and visual features, and directly outputs the target box (Deng et al. 2021; Liao et al. 2020; Yang et al. 2020), which can get rid of the computation-intensive candidate region generation, and the matching process of candidate regions and referring expression in the two-stage paradigm.

Conditional Computation

Conditional computation (also known as dynamic execution) in computer vision aims to dynamically select a subset of convolutional neural networks (CNN) to execute conditioned on the input images. Compared to the fixed inference network, it is more computation-efficient because only a part of the CNN is used. Lots of works related to conditional computation have been studied. According to the granularity of selected modules, these works can be divided into layer selection (Veit and Belongie 2018; Wang et al. 2018), filter selection (Chen et al. 2019b; Bejnordi, Blankevoort, and Welling 2019; Chen et al. 2019a; Gao et al. 2018; Herrmann, Bowen, and Zabih 2020) and layer-filter joint selection (Xia et al. 2021). Unlike the above methods, which make selection decisions based on the input images and gradually choose/skip layers or filters in the feed-forward inference, the proposed method selects layers and filters to construct the language-adaptive subnet based on a given referring expression before the feed-forward process.

Approach

In this section, we first describe the Language Adaptive Dynamic Subnets (LADS) framework. Then we introduce the expression-adaptive subnet selection based on the Gating Network, including the relevance score estimation and binary gate generation. After that, the mutual information between expressions and subnets is demonstrated. Finally, the training objectives used for the framework are described.

Framework

To achieve efficient reasoning while maintaining high accuracy, we propose the Language Adaptive Dynamic Subnets (LADS) framework, as illustrated in Fig. 2. The LADS framework consists of a REC supernet and a Gating Network. During inference, the Gating Network first extracts an expression-adaptive subnet. The subnet is then used for referring grounding.

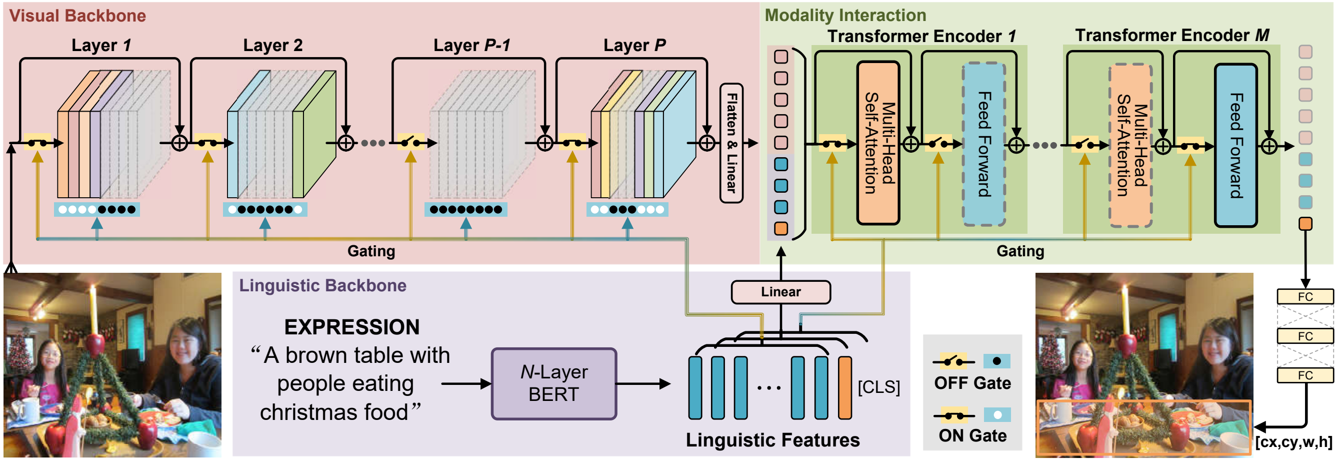

Figure 2: The overall architecture of our proposed LADS framework, which contains a REC supernet and a Gating Network. The REC supernet is composed of three components: (1) Visual Backbone, which extracts and flattens visual features from the input image, (2) Linguistic Backbone, which tokenizes referring expressions and extracts linguistic features with a prepended [CLS] token, and (3) Modality Interaction Module, which fuses the two modal features and regresses the coordinates of the referred objects. The Gating Network is the core module of the LADS framework, which takes linguistic features as input and extracts compact expression-adaptive subnets from the REC supernet by generating binary gates for the layers and filters.

The REC supernet contains three components, *i.e.* linguistic backbone, visual backbone, and modality interaction module. Given a referring expression, the N-layer BERT-based linguistic backbone (Devlin et al. 2018) tokenizes the expressions, prepends a [CLS] token, and extracts d_l-dimension linguistic features $F_l \in \mathbb{R}^{L\times d_l}$ with L tokens. Next, given an image, the visual backbone extracts C-dimension visual features $F_v \in \mathbb{R}^{C\times H\times W}$ with height H and width W. Then, the visual features F_v are flattened along the spatial dimension to match the valid input dimension of the modality interaction module and added with the sinusoidal positional embeddings to retain the spatial location information. After that, the linearly projected F_v and F_l are concatenated and fed to the modality interaction module with M transformer encoder layers for cross-modal interaction. Finally, we pass the fused features corresponding to the [CLS] token to a 3-layer fully connected (FC) layers, followed by the $Sigmoid$ function to predict the referred bounding box $b = [cx, cy, w, h]$, where (cx, cy) and (w, h) denote the center position and size, respectively.

The Gating Network, as the core module of the LADS framework, generates expression-adaptive binary gates based on the linguistic features F_l. Using these binary gates, a compact subnet dedicated to that expression can be extracted from the REC supernet. To obtain more flexible and compact subnets, we implement bi-level selection, *i.e.* layers and filters. Concretely, the residual layers and convolution filters are selected in the visual backbone, and the multi-head self-attention layers and feed-forward layers are selected in the modality interaction module. The detailed structure of Gating Network is illustrated in Fig. 3. The Gating Network can be functionally divided into two stages, namely Relevance Score Estimation and Binary Gate Generation, which will be introduced in detail next.

Relevance Score Estimation

Different from the previous conditional computation methods, such as GaterNet (Chen et al. 2019b) and AIG (Veit and Belongie 2018), which calculate the relevance score with visual features as input, in our LADS framework, the relevance scores are calculated directly according to the linguistic features. The calculation can be divided into two steps: (1) pooling the linguistic features for each candidate layer, and (2) mapping the pooled features to relevance scores.

Although there are many token feature pooling methods, such as mean pooling or using [CLS] features, they may not be suitable for our framework because different candidate layers may prefer different tokens. Therefore, we propose an attention-based parametric pooling method. Concretely, we assign a learnable layer-specific embedding $e_i \in \mathbb{R}^{d_l}$ to each candidate layer i. The token-wise attention weights $\alpha_i \in [0, 1]^L$ can be derived by calculating the inner product of e_i and F_l, followed by $Softmax$ normalization. Then, the pooled feature $h_i \in \mathbb{R}^{d_l}$ is set to the weighted sum of α_i and F_l. The detailed calculation can be denoted as:

$$\alpha_i = Softmax([e_i \cdot F_l^1, e_i \cdot F_l^2, \cdots, e_i \cdot F_l^L]) \tag{1}$$

$$h_i = \sum_{j=1}^{L} \alpha_i^j F_l^j \tag{2}$$

Finally, we utilize two fully-connected layers (FC) to calculate the relevance score r_L^i and r_C^i for the i-th layer and filters separately, which can be indicated as:

$$r_L^i = W_2^i \delta(W_1^i h_i),\ r_C^i = W_3^i \delta(W_1^i h_i) \tag{3}$$

where shared $W_1^i \in \mathbb{R}^{d_l\times d_h}$ is firstly used to reducing the dimension to d_h. $W_2^i \in \mathbb{R}^{1\times d_h}$ and $W_3^i \in \mathbb{R}^{c_i\times d_h}$ are used to calculate the relevance scores, and c_i is the number of filters. δ refers to the $GeLU$ activation function. Since the multi-head self-attention layer and feed-forward layer have the residual connection, the relevance scores are calculated the same as r_L^i. The Gating Network only contains 0.6M parameters, which is much less than the parameter amount of the whole model.

Binary Gate Generation

After estimating the relevance between layers and linguistic features, the binary gates are calculated for extracting expression-adaptive subnets from the REC supernet.

To mitigate model collapse (Veit and Belongie 2018) and maintain differentiability, we leverage the Gumbel-Softmax trick (Jang, Gu, and Poole 2016; Maddison, Mnih, and Teh 2016), and simplify it by only calculating the logits that gates are turned on, considering the particularity of the binary decision. Correspondingly, we add Logistic noises to the relevance scores instead of the Gumbel noises. In addition, inspired by the Improved Semantic Hashing (Kaiser and Bengio 2018), we randomly use the soft and hard gates during training to mitigate the gradient mismatch caused by the straight-through estimation of the hard gates. Taking the layer gate $g_L^i \in \mathbb{R}$ of the i-th layer as an example, which is calculated as follows:

$$g_s^i = Sigmoid(\hat{r}_L^i),\ g_h^i = \mathcal{I}(\hat{r}_L^i \geq 0) \tag{4}$$

$$g_L^i = \mathcal{I}(n_s \geq 0.5) \cdot g_s^i + \mathcal{I}(n_s < 0.5) \cdot g_h^i \tag{5}$$

where $\hat{r}_L^i = r_L^i + \epsilon$ is the noisy version of r_L^i, and $\epsilon \sim Logistic(0, 1)$ represents Logistic noise, which is the difference of two Gumbel i.i.d noises. $g_s^i \in [0, 1]$ and $g_h^i \in \{0, 1\}$ indicate the soft and hard gates, respectively. $\mathcal{I}(\cdot)$ is the indicative function. $n_s \sim Uniform(0, 1)$ is used to randomly select the hard or soft gates in each training batch with a probability 0.5. The channel gates $g_C^i \in \mathbb{R}^{c_i}$ can be calculated in the same way by replacing r_L^i with r_C^i.

The bottleneck layers in the visual backbone, and multi-head self-attention layers and feed-forward layers in the modality interaction module are all residual structures and can be represented as $x_{i+1} = F(x_i) + x_i$. $F(\cdot)$ and x_i denote the residual function and input feature, respectively. The layer selection can be achieved by multiplying $F(x_i)$ by layer gate g_L^i:

$$x_{i+1} = F(x_i) \cdot g_L^i + x_i \tag{6}$$

Similarly, the filters in the visual backbone can be selected by multiplying the visual feature y_i by filter gates g_C^i:

$$y_i' = y_i \cdot g_C^i \tag{7}$$

where y_i' represents the modified visual feature after filter selection. The value of one entire channel is set to 0 when the corresponding filter gate is 0.

Mutual Information between Expressions and Subnets

To encourage compact subnets and enforce the alignment between the expressions and corresponding subnets, we calculate and maximize the mutual information between the representations of expression and subnet. Considering that the expression can be represented by the linguistic features F_l, and the subnet is uniquely determined by the binary gate set $G = \{g_L, g_C\}$, the mutual information can thus be represented as:

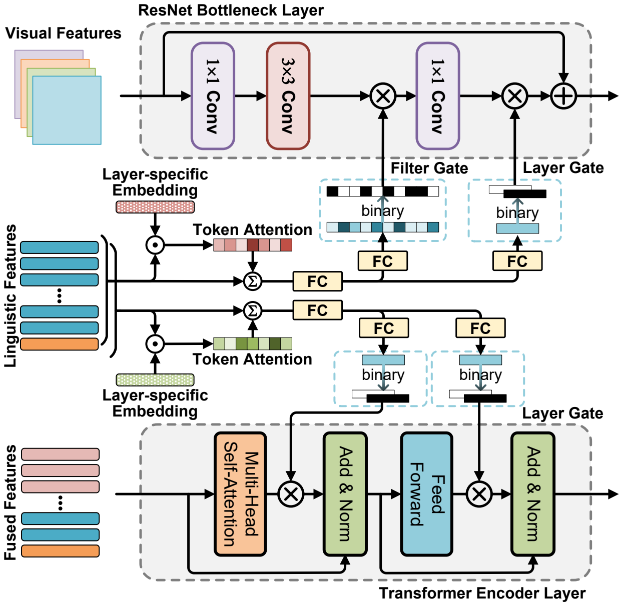

Figure 3: The detailed architecture for gate generation with linguistic features as input. The upper part of this figure shows the gate generation for one ResNet bottleneck layer in the visual backbone. The lower part shows the gate generation for one transformer encoder layer in the modality interaction module.

$$I(G; F_l) = \mathcal{H}(G) - \mathcal{H}(G|F_l) \tag{8}$$

where $\mathcal{H}(p) = -\sum_{i=1}^{N} p_i log(p_i)$ denotes the entropy of one gate. In our LADS framework, considering that each gate contains only two states (*i.e.* ”ON” and ”OFF”), the gate entropy can be simplified as $\mathcal{H}(p) = -plog(p) - (1-p)log(1-p)$, where p specifies the probability that the gate is ”ON”. We set the conditional probability $p(g|F_l) = g_s$, where g_s is the soft gate mentioned in Eq. (4), and the gate probability is set to $p(g) \approx \frac{1}{B}\sum_{i=1}^{B} p(g|F_l^i)$, which is evaluated in each training mini-batch.

By maximizing the mutual information $I(G; F_l)$, the two terms, *i.e.* gate entropy $\mathcal{H}(G)$ and conditional entropy $\mathcal{H}(G|F_l)$ are maximized and minimized, respectively. Apart from enforcing the alignment between expressions and subnets, there are two additional considerations for adopting $I(G; F_l)$: (1) Maximizing $\mathcal{H}(G)$ forces the framework to turn on/off the gates evenly, which can avoid the self-reinforcing problem (Shazeer et al. 2017) in the training process. The self-reinforcing usually exists in the conditional computation methods, and it can damage the supernet by excluding some layers or filters with zero selection probability in the early training process. Benefiting from the $\mathcal{H}(G)$, the self-reinforcing problem can be overcome in our LADS

Method	Visual	RefCOCO			RefCOCO+			RefCOCOg		Referit	Time
	Backbone	val	testA	testB	val	testA	testB	val	test	test	(ms)
MAttNet (Yu et al. 2018)	RN101	76.40	80.43	69.28	64.93	70.26	56.00	66.67	67.01	-	325
FAOA (Yang et al. 2019)	DN53	72.54	74.35	68.50	56.81	60.23	49.60	61.33	60.36	60.67	39
RCCF (Liao et al. 2020)	DLA34	-	81.06	71.85	-	70.35	56.32	-	67.01	63.79	25
MCN (Luo et al. 2020)	DN53	80.08	82.29	74.98	67.16	72.86	57.31	66.46	66.01	-	51
ReSC (Yang et al. 2020)	DN53	77.63	80.45	72.30	63.59	68.36	56.81	67.30	67.20	64.60	53
TransVG (Deng et al. 2021)	RN50	80.32	82.67	78.12	63.50	68.15	55.63	67.66	67.44	69.76	41
Ref-NMS (Chen et al. 2021)	RN101	80.70	84.00	76.04	68.25	73.68	59.42	70.55	70.62	-	-
LADS (ours)	RN50	**82.85**	**86.67**	**78.57**	**71.16**	**77.64**	**59.82**	**71.56**	**71.66**	**71.08**	**20**
ViLBERT (Lu et al. 2019)	RN101	-	-	-	72.34	78.52	62.61	-	-	-	400
UNITER_L (Chen et al. 2020)	RN101	81.41	87.04	74.17	75.90	81.45	66.70	74.86	75.77	-	416
VILLA_L (Gan et al. 2020)	RN101	82.39	87.48	74.84	76.17	81.54	66.84	76.18	76.71	-	417
MDETR (Kamath et al. 2021)	RN101	86.75	89.58	81.41	79.52	84.09	70.62	81.64	80.89	-	65
LADS (ours)	RN50	**87.80**	**91.23**	**84.03**	**79.65**	**84.86**	**71.97**	**82.67**	**81.96**	**78.82**	**18**

Table 1: Comparisons with state-of-the-art methods on the RefCOCO, RefCOCO+, RefCOCOg, and Referit datasets. RN101, RN50, and DN53 are shorthand for the ResNet101, ResNet50, and DarkNet53, respectively. The inference time is tested on 1080 Ti GPU and averaged over all referring expressions.

framework by preventing the gates from always turning on or off, and the actual network capacity and accuracy can be kept without layer/filter dropping. (2) Minimizing $\mathcal{H}(G|F_l)$ constraints the framework to generate unambiguous gates for each sample conditioned on the specific referring expression. In addition, it can also push the soft gates to hard gates in Eq. (5) by optimizing the gate probability to 1 or 0, and the gap between training and evaluation is narrowed.

Training Objectives

The LADS framework is trained end-to-end, and similar to DETR (Carion et al. 2020) and TransVG (Deng et al. 2021), the L1 loss and Generalized IoU (GIoU) loss are used between the predicted referring box $b = (cx, cy, w, h)$ and ground truth bounding box $\hat{b} = (\hat{cx}, \hat{cy}, \hat{w}, \hat{h})$. The negative mutual information $-I(G; F_l)$ in Eq. (8) is also included to constrain the alignment between the referring expressions and the corresponding REC subnets. The total training loss can be summarized as:

$$\mathcal{L}_{total} = \mathcal{L}_{L1}(b, \hat{b}) + \lambda_{giou}\mathcal{L}_{giou}(b, \hat{b}) - \lambda_{mi}I(G; F_l) \quad (9)$$

The λ_{giou} and λ_{mi} are coefficients for the GIoU loss and negative mutual information loss, which are set to 1.0 and 0.1 in the experiments, respectively.

Experiments

In this section, we conduct experiments on the proposed Language Adaptive Dynamic Subnets (LADS) framework to evaluate the accuracy and efficiency on REC datasets. The dynamicity of the REC subnets and the alignment between subnets and expressions are also evaluated.

Experimental Setting

Datasets There are five REC datasets used in the experiments, including RefCOCO (Yu et al. 2016), RefCOCO+ (Yu et al. 2016), RefCOCOg (Mao et al. 2016), Referit (Kazemzadeh et al. 2014) and large-scale pre-training dataset. RefCOCO and RefCOCO+, which are officially split into train, val, testA, and testB sets, have 19,994 images with 142,210 referring expressions and 19,992 images with 141,564 referring expressions, respectively. RefCOCOg (Nagaraja, Morariu, and Davis 2016) has 25,799 images with 95,010 referring expressions, which is officially split into train, val, and test sets. Referit has 20,000 images collected from SAIAPR-12 (Escalante et al. 2010), which is split into train and test sets. The large-scale pre-training dataset has 174k images with approximately 6.1M distinct referring expressions, which contains the train sets of RefCOCO/+/g, Referit, VG regions (Krishna et al. 2017), and Flickr entities (Plummer et al. 2015).

Evaluation Metric We use *Prec*@0.5 evaluation protocol to evaluate the accuracy. Given a referring expression, a predicted region is considered correct if its intersection-over-union (IoU) with the ground-truth bounding box is greater than 0.5. In addition, we also report the average time taken for a complete inference of the LADS framework.

Implementation Details

Training All models are trained on the NVIDIA A100 GPU with CUDA 11.4. For the visual backbone, we use ResNet50 (He et al. 2016) pre-trained on MSCOCO (Lin et al. 2014), where overlapping images in the val/test sets of the corresponding datasets are excluded. For the linguistic backbone, we use the first six layers of BERT (Devlin et al. 2018) provided by HuggingFace (Wolf et al. 2020). The rest of the model is initialized using Xavier initialization (Glorot and Bengio 2010). The input images are resized to 512×512, and the max expression length is 40. All models are end-to-end optimized by AdamW (Loshchilov and Hutter 2020) optimizer with weight decay of 1e-4. The initial learning rate of visual backbone and linguistic backbone is 1e-5, and the initial learning rate of the rest is 1e-4. We train for 120 epochs with a batch size of 256, where the learning rate is reduced

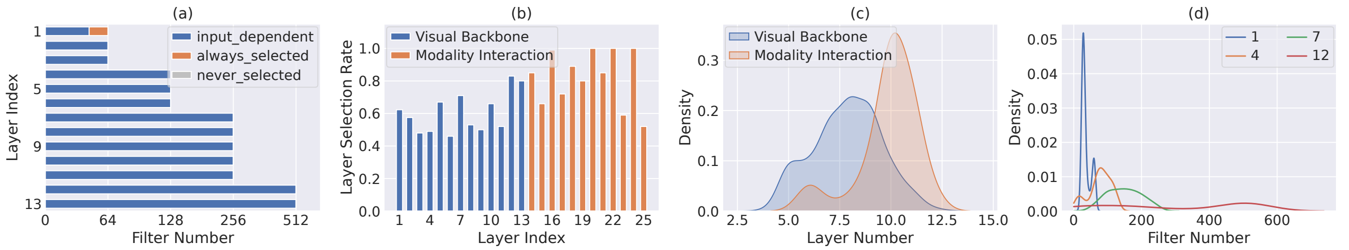

Figure 4: Dynamicity of the REC subnets. From left to right: (a) dynamicity of the filter selection, (b) dynamicity of the layer selection, (c) distribution over the number of selected layers, and (d) distribution over the number of selected filters.

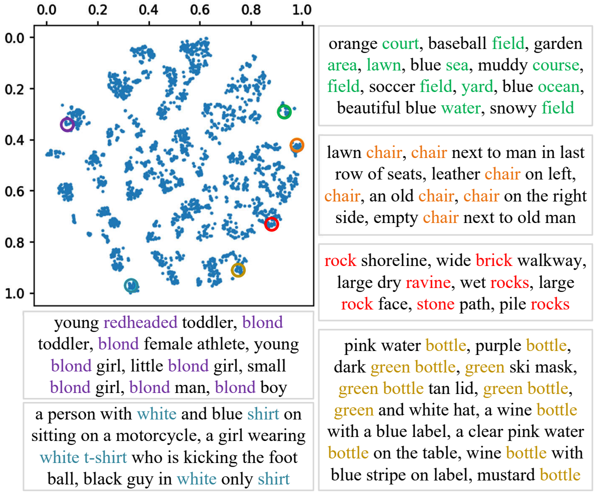

Figure 5: t-SNE visualization of the gate sets, *i.e.* the subnet architectures. Six regions are circled and the corresponding phrases are shown around the figure. Best viewed in color.

by 10 after 90 epochs. In large-scale pre-training and fine-tuning, we train for 40 and 20 epochs with batch sizes of 512 and 256, where the learning rate is reduced by 10 after 30 and 10 epochs, respectively. Following the common practice in (Deng et al. 2021; Yang et al. 2020), we perform data augmentation at the training stage, including random resize, random crop, and horizontal flip.

Evaluation and Inference In the evaluation and inference stage, there are two differences from training. First, the Logistic noise ϵ in Eq. (4) is set to 0, and there is no randomness in the model. Second, the hard gates g_h^i in Eq. (5) are always used, which means that the layers and filters are directly removed when $g_h^i = 0$ and the rest components with $g_h^i = 1$ make up the expression-adaptive subnets.

Comparison with State-of-the-art Methods

To estimate the effectiveness of the proposed LADS framework, we conduct quantitative experiments on four widely used datasets, *i.e.* RefCOCO (Yu et al. 2016), RefCOCO+ (Yu et al. 2016), RefCOCOg (Mao et al. 2016), and Referit (Kazemzadeh et al. 2014), and compare results with state-of-the-art methods. The main results are summarized to Table 1. These methods can be split into two settings, *i.e.* models trained on each dataset and models pre-trained on large-scale datasets followed by fine-tuning. The proposed LADS framework achieves the best accuracy and fastest inference speed in both settings. Compared to the SOTA method Ref-NMS (Chen et al. 2021), LADS has a better $Prec@0.5$ with +2.15%/ +2.67%/ +2.53% on RefCOCO, +2.91%/ +3.96%/ +0.40%/ on RefCOCO+, and +1.01%/ +1.04%/ on RefCOCOg. Compared to the transformer-based method TransVG (Deng et al. 2021), our method also has better performance with +2.53%/ +4.00%/ +0.45% on RefCOCO, +7.66%/ +9.49%/ +4.19%/ on RefCOCO+, +3.90%/ +4.22%/ on RefCOCOg, and +1.32% on Referit. Our model also runs the fastest with only 20ms inference time.

Inspired by recent large-scale pre-training approaches, we also pre-train our method on the large-scale dataset, similar to MDETR (Kamath et al. 2021), and then fine-tune it on the REC datasets. Our LADS also outperforms MDETR and has a large accuracy gap compared to the non-pretrained methods, suggesting that large-scale data is necessary for better performance due to the complexity of REC. Interestingly, our pre-trained model has a 2ms speedup from 20ms to 18ms, which means that better alignment of expressions and subnets can be constrained with large-scale data, resulting in more compact REC subnets.

Dynamicity of the REC Subnets

To verify the dynamicity of the expression-adaptive REC subnets, we count the selection rate of filters and layers on the test set of Referit. In addition, the expression-wise numbers of selected layers and filters are also counted. These statistics are illustrated in Fig. 4.

Fig. 4 (a) shows the dynamicity of the filter selection. It can be observed that almost all the filters are highly input-dependent, which means our method can select relevant filters conditioned on various referring expressions. There is no never-selected filter on all layers, which means the self-reinforcing problem can be avoided, and thus the model capacity can be preserved. Fig. 4 (b) shows the dynamicity of the layer selection. It can be seen that both layers in the visual backbone and modality interaction module have high expression adaptability. The layers of visual backbone are chosen by nearly half. The selection rate of MHSA and FFN in the modality interaction module varies, which means the contributions of the two types of layers are different.

Fig. 4 (c) shows the distribution over the number of selected layers. The visual backbone of the dynamic subnets

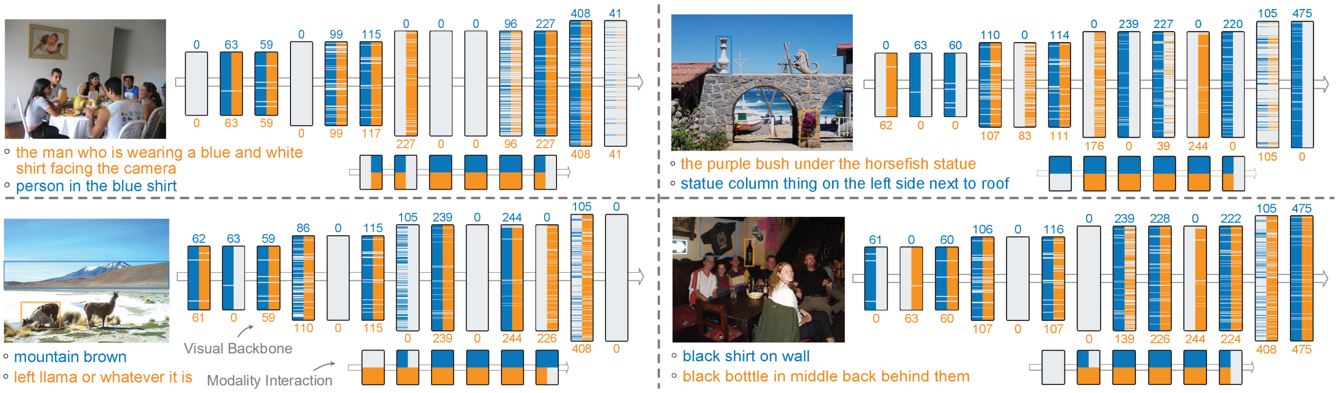

Figure 6: Structures of the expression-adaptive REC subnets. The dynamic visual backbone and modality interaction module are visualized. Selected and unselected filters/layers are marked with color and gray, respectively. The number of selected filters in the four sets of layers with dimensions of 64, 128, 256, and 512 is displayed around the Visual Backbone.

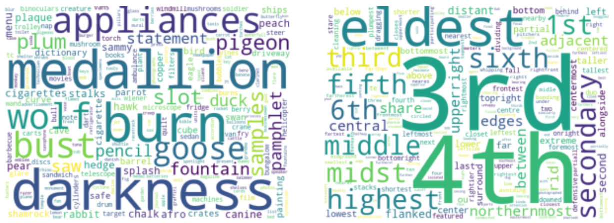

Figure 7: WordCloud of words assigned to the visual backbone (left) and modality interaction module (right). The visual backbone prefers categories and attributes, whereas the modality interaction module prefers context.

has 8 layers on average and 4 layers at least, which means very shallow visual backbones can be extracted conditioned on the referring expression. The modality interaction module of the dynamic subnets has 10 layers on average and 6 layers for some expressions. Fig. 4 (d) shows the distribution over the number of selected filters. For simplicity, only 4 layers of filters are visualized. It can be observed that the selected filters vary with different referring expressions. Moreover, as the number of filters increases, its variance increases, which means the filters of deeper layers in the visual backbone are stronger dynamics.

In addition, we also show several specific structures of the expression-adaptive REC subnets in Fig. 6. It can be seen that the LADS framework can extract subnets with different depth/layers and width/channels for various referring expressions. The structures of the REC subnets seem to be similar when the corresponding referring expressions have similar semantic information.

Alignment of Expressions and Subnets

The LADS framework is designed to generate compact REC subnets for any referring expressions, and the architectures of the subnets are desired to align with the expressions, *i.e.* the structures of the subnets corresponding to semantically similar expressions should also be similar.

Since the architectures of subnets can be uniquely determined by the binary gate set $G = \{g_L, g_C\}$, we project the high-dimensional G to the 2-d coordinates for visualization by using t-SNE. The result is shown in Fig. 5. Due to the uncountable expressions, there is no obvious clustering, which is different from the classification tasks (*e.g.* ImageNet and CIFAR) with a limited number of categories. We visualize the referring expressions corresponding to the architectures in six regions and find that similar architectures do correspond to semantically similar expressions, showing that the expressions and REC subnets are well aligned.

We also visualize the words assigned to the visual backbone and modality interaction module to perform layer-granularity alignment inspections in Fig. 7. Concretely, we first calculate the word score $s_w \in [0,1]^{(P+M)}$ for word w by averaging the α_i derived from Eq. (1) over all expressions, where P and M represent the number of candidate layers in visual backbone and modality interaction module, respectively. Then we normalize s_w to $\hat{s}_w$ using $Softmax$. Finally, the scores of word w assigned to the visual backbone and modality interaction module are set to the sum of the first P elements and the last M elements of $\hat{s}_w$, respectively. Interestingly, the visual backbone prefers categories and attributes (*e.g.* ”medallion” and ”darkness”), whereas the modality interaction module prefers context (*e.g.* ”3rd”, ”eldest” and ”middle”).

Conclusions and Future Works

In this paper, we propose a neat yet efficient framework named Language Adaptive Dynamic Subnets (LADS), which can adaptively extract REC subnets conditioned on the referring expressions. Extensive experiments show that LADS achieves superior inference speed and accuracy compared to state-of-the-art methods, indicating the LADS framework’s effectiveness and prospects for more economical and faster REC inference. For future studies, we think there are two directions worth pursuing, *i.e.* (1) study the correspondence between language and subnets to achieve more discriminative and accurate subnet selection, and (2) combine the image-dependent selection mechanism to realize joint selection based on expressions and images.

Acknowledgments

This work is supported in part by National Key Research and Development Program of China under Grant 2020AAA0107400, Zhejiang Provincial Natural Science Foundation of China under Grant LR19F020004, National Natural Science Foundation of China under Grant U20A20222, National Science Foundation for Distinguished Young Scholars under Grant 62225605, Ant Group, and sponsored by CAAI-HUAWEI MindSpore Open Fund.

References

Bajaj, M.; Wang, L.; and Sigal, L. 2019. G3raphground: Graph-based language grounding. In *Int. Conf. Comput. Vis.*

Bejnordi, B. E.; Blankevoort, T.; and Welling, M. 2019. Batch-shaping for learning conditional channel gated networks. *arXiv preprint arXiv:1907.06627*.

Cai, H.; Gan, C.; Wang, T.; Zhang, Z.; and Han, S. 2019. Once-for-All: Train One Network and Specialize it for Efficient Deployment. In *Int. Conf. Learn. Represent.*

Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; and Zagoruyko, S. 2020. End-to-end object detection with transformers. In *Eur. Conf. Comput. Vis.*, 213–229.

Chen, J.; Zhu, Z.; Li, C.; and Zhao, Y. 2019a. Self-adaptive network pruning. In *Adv. Neural Inform. Process. Syst.*, 175–186.

Chen, L.; Ma, W.; Xiao, J.; Zhang, H.; and Chang, S.-F. 2021. Ref-NMS: breaking proposal bottlenecks in two-stage referring expression grounding. In *AAAI*.

Chen, Y.-C.; Li, L.; Yu, L.; El Kholy, A.; Ahmed, F.; Gan, Z.; Cheng, Y.; and Liu, J. 2020. Uniter: Universal image-text representation learning. In *Eur. Conf. Comput. Vis.*, 104–120.

Chen, Z.; Li, Y.; Bengio, S.; and Si, S. 2019b. You look twice: Gaternet for dynamic filter selection in cnns. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 9172–9180.

Deng, J.; Yang, Z.; Chen, T.; Zhou, W.; and Li, H. 2021. Transvg: End-to-end visual grounding with transformers. In *Int. Conf. Comput. Vis.*

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Escalante, H. J.; Hernández, C. A.; Gonzalez, J. A.; López-López, A.; Montes, M.; Morales, E. F.; Sucar, L. E.; Villasenor, L.; and Grubinger, M. 2010. The segmented and annotated IAPR TC-12 benchmark. *CVIU*.

Gan, Z.; Chen, Y.-C.; Li, L.; Zhu, C.; Cheng, Y.; and Liu, J. 2020. Large-scale adversarial training for vision-and-language representation learning. *Advances in Neural Information Processing Systems*, 33: 6616–6628.

Gao, X.; Zhao, Y.; Dudziak, Ł.; Mullins, R.; and Xu, C.-z. 2018. Dynamic channel pruning: Feature boosting and suppression. *arXiv preprint arXiv:1810.05331*.

Glorot, X.; and Bengio, Y. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*.

He, K.; Gkioxari, G.; Dollár, P.; and Girshick, R. 2017. Mask r-cnn. In *IEEE Conf. Comput. Vis. Pattern Recog.*

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *IEEE Conf. Comput. Vis. Pattern Recog.*

Herrmann, C.; Bowen, R. S.; and Zabih, R. 2020. Channel selection using gumbel softmax. In *Eur. Conf. Comput. Vis.*, 241–257.

Hong, R.; Liu, D.; Mo, X.; He, X.; and Zhang, H. 2019. Learning to compose and reason with language tree structures for visual grounding. *IEEE Trans. Pattern Anal. Mach. Intell.*

Jang, E.; Gu, S.; and Poole, B. 2016. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*.

Kaiser, Ł.; and Bengio, S. 2018. Discrete autoencoders for sequence models. *arXiv preprint arXiv:1801.09797*.

Kamath, A.; Singh, M.; LeCun, Y.; Synnaeve, G.; Misra, I.; and Carion, N. 2021. MDETR-modulated detection for end-to-end multi-modal understanding. In *Int. Conf. Comput. Vis.*, 1780–1790.

Kazemzadeh, S.; Ordonez, V.; Matten, M.; and Berg, T. 2014. Referitgame: Referring to objects in photographs of natural scenes. In *EMNLP*.

Krishna, R.; Zhu, Y.; Groth, O.; Johnson, J.; Hata, K.; Kravitz, J.; Chen, S.; Kalantidis, Y.; Li, L.-J.; Shamma, D. A.; et al. 2017. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *Int. J. Comput. Vis.*

Li, Q.; Fu, J.; Yu, D.; Mei, T.; and Luo, J. 2018. Tell-and-answer: Towards explainable visual question answering using attributes and captions. *EMNLP*.

Liao, Y.; Liu, S.; Li, G.; Wang, F.; Chen, Y.; Qian, C.; and Li, B. 2020. A real-time cross-modality correlation filtering method for referring expression comprehension. In *IEEE Conf. Comput. Vis. Pattern Recog.*

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *Eur. Conf. Comput. Vis.*

Liu, D.; Zhang, H.; Wu, F.; and Zha, Z.-J. 2019. Learning to assemble neural module tree networks for visual grounding. In *Int. Conf. Comput. Vis.*

Liu, J.; Wang, L.; and Yang, M.-H. 2017. Referring expression generation and comprehension via attributes. In *Int. Conf. Comput. Vis.*

Loshchilov, I.; and Hutter, F. 2020. Decoupled weight decay regularization. In *Int. Conf. Learn. Represent.*

Lu, J.; Batra, D.; Parikh, D.; and Lee, S. 2019. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *Advances in neural information processing systems*, 32.

Luo, G.; Zhou, Y.; Sun, X.; Cao, L.; Wu, C.; Deng, C.; and Ji, R. 2020. Multi-task collaborative network for joint referring expression comprehension and segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*

Maddison, C. J.; Mnih, A.; and Teh, Y. W. 2016. The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*.

Mao, J.; Huang, J.; Toshev, A.; Camburu, O.; Yuille, A. L.; and Murphy, K. 2016. Generation and comprehension of unambiguous object descriptions. In *IEEE Conf. Comput. Vis. Pattern Recog.*

Nagaraja, V. K.; Morariu, V. I.; and Davis, L. S. 2016. Modeling context between objects for referring expression understanding. In *Eur. Conf. Comput. Vis.*

Plummer, B. A.; Wang, L.; Cervantes, C. M.; Caicedo, J. C.; Hockenmaier, J.; and Lazebnik, S. 2015. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *Int. Conf. Comput. Vis.*

Ren, S.; He, K.; Girshick, R.; and Sun, J. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. *Adv. Neural Inform. Process. Syst.*

Salvador, A.; Giró-i Nieto, X.; Marqués, F.; and Satoh, S. 2016. Faster r-cnn features for instance search. In *IEEE Conf. Comput. Vis. Pattern Recog.*

Shazeer, N.; Mirhoseini, A.; Maziarz, K.; Davis, A.; Le, Q.; Hinton, G.; and Dean, J. 2017. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv preprint arXiv:1701.06538*.

Veit, A.; and Belongie, S. 2018. Convolutional networks with adaptive inference graphs. In *Eur. Conf. Comput. Vis.*, 3–18.

Wang, P.; Wu, Q.; Cao, J.; Shen, C.; Gao, L.; and Hengel, A. v. d. 2019. Neighbourhood watch: Referring expression comprehension via language-guided graph attention networks. In *IEEE Conf. Comput. Vis. Pattern Recog.*

Wang, X.; Yu, F.; Dou, Z.-Y.; Darrell, T.; and Gonzalez, J. E. 2018. Skipnet: Learning dynamic routing in convolutional networks. In *Eur. Conf. Comput. Vis.*, 409–424.

Wolf, T.; Debut, L.; Sanh, V.; Chaumond, J.; Delangue, C.; Moi, A.; Cistac, P.; Rault, T.; Louf, R.; Funtowicz, M.; et al. 2020. Transformers: State-of-the-art natural language processing. In *EMNLP*.

Xia, W.; Yin, H.; Dai, X.; and Jha, N. K. 2021. Fully dynamic inference with deep neural networks. *IEEE Transactions on Emerging Topics in Computing*.

Yang, S.; Li, G.; and Yu, Y. 2019. Cross-modal relationship inference for grounding referring expressions. In *IEEE Conf. Comput. Vis. Pattern Recog.*

Yang, Z.; Chen, T.; Wang, L.; and Luo, J. 2020. Improving one-stage visual grounding by recursive sub-query construction. In *Eur. Conf. Comput. Vis.*

Yang, Z.; Gong, B.; Wang, L.; Huang, W.; Yu, D.; and Luo, J. 2019. A fast and accurate one-stage approach to visual grounding. In *Int. Conf. Comput. Vis.*, 4683–4693.

Yu, L.; Lin, Z.; Shen, X.; Yang, J.; Lu, X.; Bansal, M.; and Berg, T. L. 2018. Mattnet: Modular attention network for referring expression comprehension. In *IEEE Conf. Comput. Vis. Pattern Recog.*

Yu, L.; Poirson, P.; Yang, S.; Berg, A. C.; and Berg, T. L. 2016. Modeling context in referring expressions. In *Eur. Conf. Comput. Vis.*

Zhou, Y.; Ji, R.; Luo, G.; Sun, X.; Su, J.; Ding, X.; Lin, C.-W.; and Tian, Q. 2021. A real-time global inference network for one-stage referring expression comprehension. *IEEE Transactions on Neural Networks and Learning Systems*.