Comprehension-Guided Referring Expressions

Ruotian Luo — *TTI-Chicago* — rluo@ttic.edu

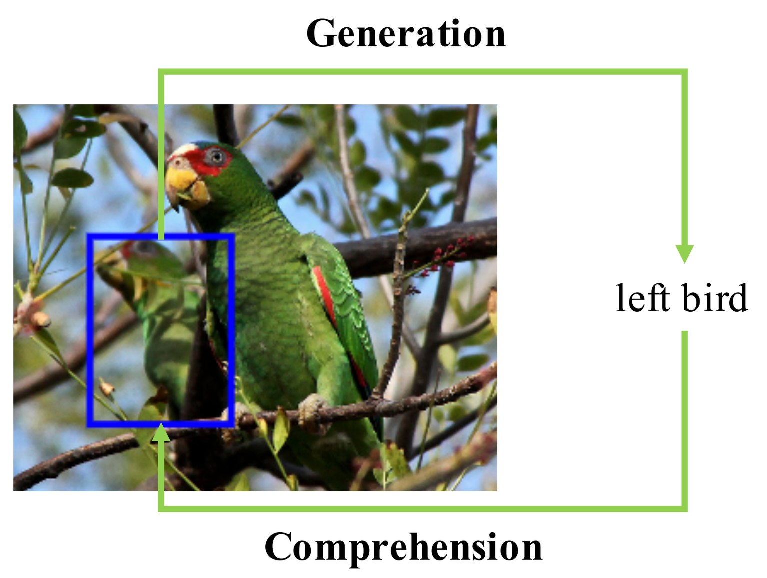
Greg Shakhnarovich — *TTI-Chicago* — greg@ttic.edu

Introduction

Referring expressions describe an object or region in the image, with the goal of identifying it uniquely to a listener.

RE Generation: generating a discriminative referring expression for an object in an image.

RE Comprehension: localizing an object in an image given a referring expression.



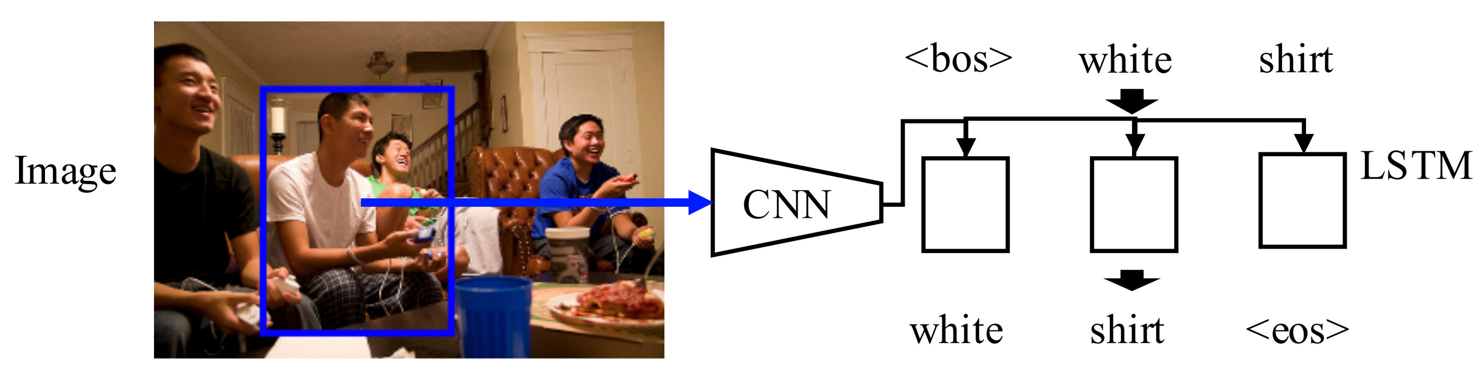
Motivation: use a standalone comprehension model to "tell" the generator how to improve the expressions it produces.

Base Models

Generation model: takes inputs of an image I and an internal region r, and outputs an expression w.

$$G : I \times r \to w$$

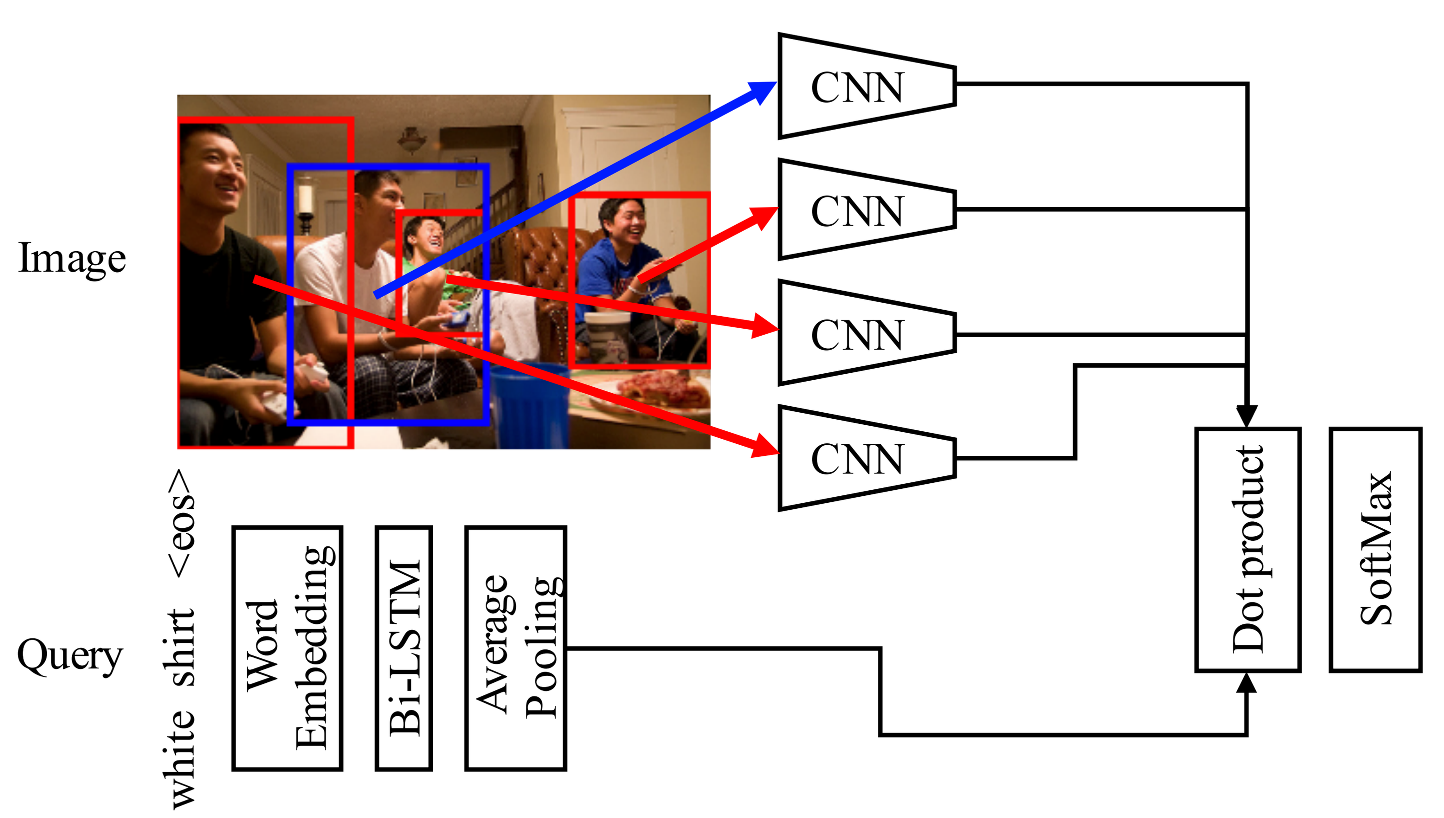
The model is a standard CNN+LSTM which is trained to maximize $P_G(w|I, r)$.



Comprehension model: The comprehension task is to select a region (bounding box) $\hat{r}$ from a set of regions $\mathcal{R} = \{r_i\}$ given a query expression q and the image I.

$$C : I \times q \times \mathcal{R} \to r,\ r \in \mathcal{R} \tag{1}$$

The model is trained to maximize $P_C(r^*|I, q, \mathcal{R})$.



Generate and rerank

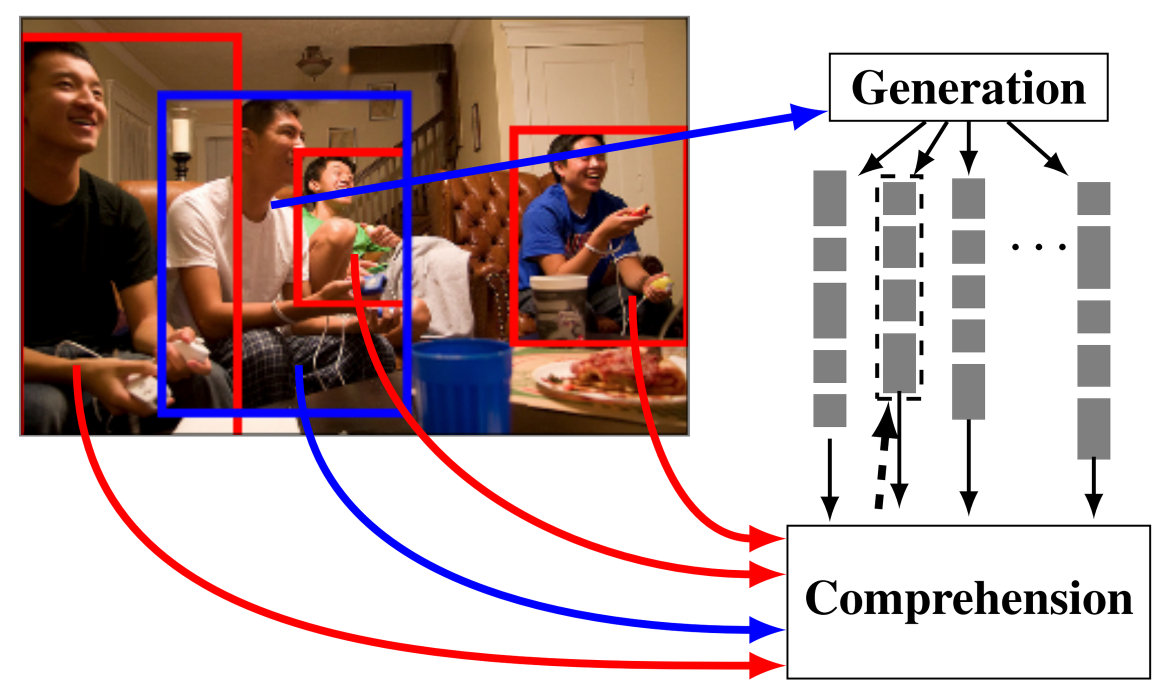
This method composes the comprehension model during test time. The pipeline is as follows:

1. Generate candidate expressions $\{w^1, \ldots, w^n\}$ according to $P_G(\cdot|I, r)$.
2. Select w^k with $k = \operatorname{argmax}_i score(w^i)$.

The score function is a weighted combination of the log perplexity and comprehension loss.

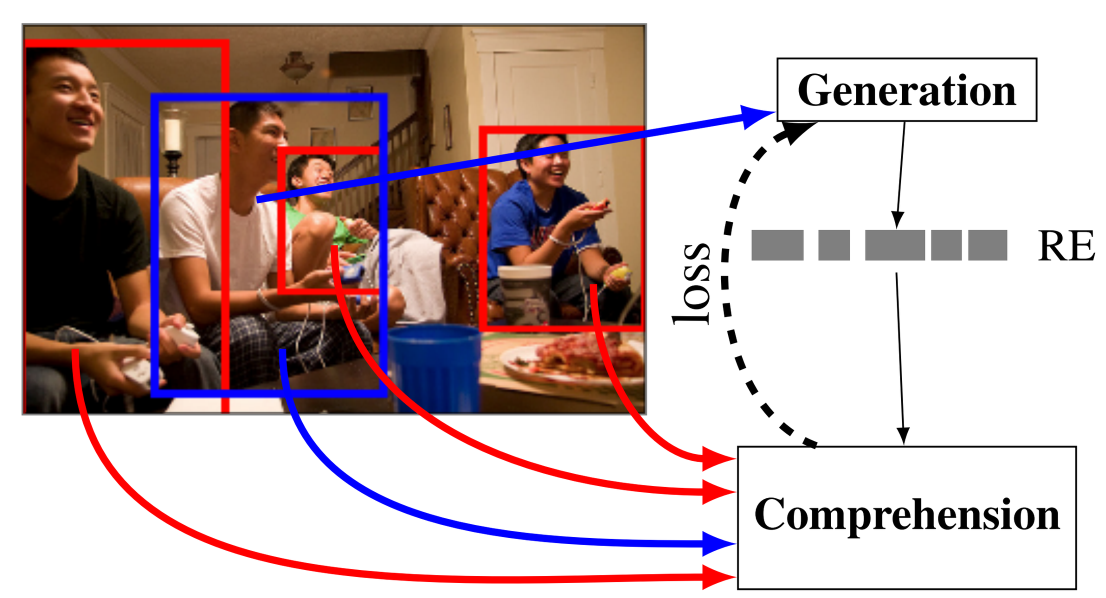
$$score(w) = \frac{1}{T}\sum_{t=1}^{T} \log p_G(w_t|r, w_{1..t-1}) + \gamma \log p_C(r|I, \mathcal{R}, w),$$

where w_k is the k-th token of w, T is the length of w.



Training by proxy

In this method the generation and comprehension model are connected and the generation model is optimized to lower discriminative comprehension loss (in addition to the cross entropy loss)



The comprehension model must correctly identify the target (blue) region based on the generated referring expression; comprehension loss (dashed) is back-propagated to update the generator.

Differentiable approximation: to be able to back-propogate, we use the softmax output of the generation model instead of the one-hot sampled output as the input of the comprehension model.

Results

Comprehension result: with the same visual feature and simpler model our model can achieve competitive results. It proves our model can provide useful signal to generation.

	RefCOCO		RefCOCO+		RefCOCOg
	Test A	Test B	Test A	Test B	Val
Baseline	63.15%	64.21%	48.73%	42.13%	55.16%
MMI	71.72%	71.09%	52.44%	47.51%	62.14%
visdif+MMI	73.98%	76.59%	59.17%	**55.62%**	64.02%
neg bag	**75.6%**	**78.0%**	-	-	**68.4%**
Ours	74.04%	73.43%	**60.26%**	55.03%	65.36%

Generation result: 'Acc' is the "comprehension accuracy" of the generated expressions according to our comprehension model. Higher 'Acc' proves the effectiveness of differentiable approximation.
Our generate-and-rerank model gets consistently better results on automatic comprehension accuracy and on fluency-based metrics like BLEU, showing benefit from comprehension-guided reranking.
CL, MSS and SMIXEC are three training schedules of training-by-proxy. They perform less well than Rerank, but they are still better than the baseline from the human evaluation result.

	RefCOCO				RefCOCO+				RefCOCOg	
	Test A		Test B		Test A		Test B		val	
	Acc	Bleu 1	Acc	Bleu 1	Acc	Bleu 1	Acc	Bleu 1	Acc	Bleu 1
Max Likelihood	74.80%	0.477	72.81%	0.553	62.10%	**0.391**	46.21%	0.331	61.96%	0.437
MMI	78.78%	0.478	74.01%	0.547	67.79%	0.370	55.21%	0.324	70.38%	0.428
CL	**80.14%**	0.4586	75.44%	0.5434	68.54%	0.3683	**55.87%**	**0.3409**	70.74%	**0.4439**
MSS	79.94%	0.4574	**75.93%**	0.5403	**69.41%**	0.3763	55.59%	0.3386	**70.80%**	0.4377
SMIXEC	79.99%	**0.4855**	75.60%	**0.5536**	69.05%	0.3847	54.71%	0.3275	70.02%	0.4338
sample	78.38%	0.5201	73.08%	0.5842	62.45%	0.3925	47.86%	0.3354	66.72%	0.4406
Rerank	**97.23%**	**0.5209**	**94.96%**	**0.5935**	**77.32%**	**0.3956**	**67.65%**	**0.3368**	**76.65%**	**0.4410**

Human evaluation results (human comprehension accuracy on generated expressions):

	RefCOCO		RefCOCO+	
	Test A	Test B	TestA	TestB
MMI	53%	61%	39%	35%
SMIXEC	62%	68%	**46%**	25%
Rerank	**66%**	**75%**	43%	**47%**

Sample results:

MLE: **person in blue**
MMI: **person in black**

CL: **left person**
MSS: **left person**
SMIXEC: **second from left**
Rerank: **second guy from left**

MLE: **left most sandwich**
MMI: **left most piece of sandwich**

CL: **left most sandwich**
MSS: **left most sandwich**
SMIXEC: **left bottom sandwich**
Rerank: **bottom left sandwich**

MLE: **hand holding the**
MMI: **hand**

CL: **hand closest to us**
MSS: **hand closest to us**
SMIXEC: **hand closest to us**
Rerank: **hand closest to us**

MLE: **giraffe with head down**
MMI: **tallest giraffe**

CL: **big giraffe**
MSS: **big giraffe**
SMIXEC: **giraffe with head up**
Rerank: **giraffe closest to us**

Conclusion

In this paper, we propose to use a learned comprehension model to guide generating better referring expressions. Our training by proxy method and generate and rerank method is shown to be promising, with the generate-and-rerank method obtaining particularly good results across datasets.