Show, Recall, and Tell: Image Captioning with Recall Mechanism

Li Wang,[1,3] **Zechen Bai,**[2,3] **Yonghua Zhang,**[3] **Hongtao Lu**[1*]
[1]Department of Computer Science and Engineering, Shanghai Jiao Tong University, China
[2]Institute of Software, Chinese Academy of Sciences, China
[3]AI-Lab Visual Search Team, Bytedance
{liwang.sjtu, ustbbzch}@gmail.com, mpeg21@hotmail.com, htlu@sjtu.edu.cn

Abstract

Generating natural and accurate descriptions in image captioning has always been a challenge. In this paper, we propose a novel recall mechanism to imitate the way human conduct captioning. There are three parts in our recall mechanism : recall unit, semantic guide (SG) and recalled-word slot (RWS). Recall unit is a text-retrieval module designed to retrieve recalled words for images. SG and RWS are designed for the best use of recalled words. SG branch can generate a recalled context, which can guide the process of generating caption. RWS branch is responsible for copying recalled words to the caption. Inspired by pointing mechanism in text summarization, we adopt a soft switch to balance the generated-word probabilities between SG and RWS. In the CIDEr optimization step, we also introduce an individual recalled-word reward (WR) to boost training. Our proposed methods (SG+RWS+WR) achieve BLEU-4 / CIDEr / SPICE scores of 36.6 / 116.9 / 21.3 with cross-entropy loss and 38.7 / 129.1 / 22.4 with CIDEr optimization on MSCOCO Karpathy test split, which surpass the results of other state-of-the-art methods.

Introduction

Image captioning is defined as automatically generating a descriptive statement from an image. This task needs to exploit image information and then to generate a natural caption. Image captioning can be applied to a wide range of domains, for example, automatically adding subtitles to images or videos, which can do great help in search task.

In the last few years, encoder-decoder models have been designed to accomplish the captioning task in many methods (Socher et al. 2014; Vinyals et al. 2015; Qin et al. 2019). The role of encoder in captioning is to extract sufficient and useful visuqinal features from the image, and image has been mostly encoded by using Convolutional Neural Network (CNNs) such as ResNet (He et al. 2016). Meanwhile, the role of decoder is to exploit semantic part from encoded visual information and then decode it word by word. Recurrent neural network (RNNs) is the most commonly used method of decoder in captioning.

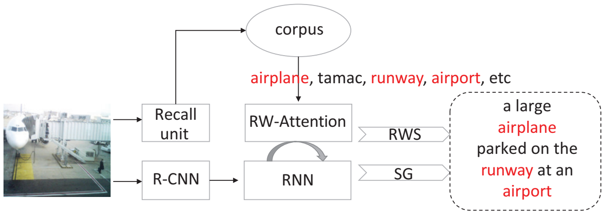

corpus
airplane, tamac, runway, airport, etc
Recall unit
RW-Attention
RWS
R-CNN
RNN
SG
a large airplane parked on the runway at an airport

Figure 1: An overview of our proposed methods. Recall unit is a text-retrieval module. RW-Attention denotes recalled-words attention. RWS represents recalled-words slot. SG is semantic guide. In short, we introduce a recall unit to the traditional captioning model, employing recalled words to boost the performance of captioning.

On one hand, visual attention methods (Xu et al. 2015; Lu et al. 2017) have brought significant improvement in captioning on the most evaluation metrics like BLEU (Papineni et al. 2002), METEOR (Denkowski and Lavie 2014) and CIDEr (Vedantam, Lawrence Zitnick, and Parikh 2015). In the process of generating caption, visual attention methods can allocate different weights to different regions of an image, which prompt model only focus on the crucial parts of the image.

On the other hand, semantic methods have also improved captioning performance remarkably. Numerous methods (You et al. 2016; Lu et al. 2018) employ detection technique to obtain words of objects and attributes, then take those words as known objects to generate caption, but these methods are highly dependent on the performance of detection model. (Mun, Cho, and Han 2016) have retrieved relevant guide texts according to the visual similarity between images, but they fail to construct a direct relationship between the guide texts and the generated caption.

Looking back the above methods of captioning, all information to generate caption is extracted from only one given image, but when a human describes an image, is all the information used only from this image? Usually not. When people are intend to do something, they will first recall past experiences, imitate them appropriately, and then do it. This is human instinct with no exception in captioning. People

[*] Corresponding author

will recall how similar images were described, and then use these similar patterns to generate a caption for the image.

For the purpose of making captioning model to describe images in the way like human beings do, in this paper, we introduce a novel recall mechanism into captioning model. In order to recall useful and relevant words for each image, we apply an image-text matching model similar to that brought by (Lee et al. 2018) as our text-retrieval model, and captions from training data are taken as our corpus. In this image-text matching model, we embed image feature and text feature into a common space, then calculate the cosine similarity between them. Triplet loss (Kiros, Salakhutdinov, and Zemel 2014b; Socher et al. 2014) is the objective function for each mini-batch during training.

To make recalled words more relevant to the image and to filter out useless words, for each image, we construct a set of recalled words just from top K captions of text-retrieval task. As illustrated in Figure 1, RW-Attention module is applied to obtain weights of recalled words, and these weights are used into two branches: semantic guide (SG), recalled-word slot (RWS), then the final caption is generated by these two branches. As above, we mainly have the following contributions in this paper:

1. In order to imitate the human behavior of recalling, we apply an image-text matching model to retrieve recalled words for each image.
2. We propose two methods to utilize recalled words: semantic guide and recalled-word slot.
3. In the CIDEr optimization stage, we propose a novel recalled-word reward to boost caption performance.

We evaluate the performance of our proposed methods on MSCOCO Karpathy test split with both cross-entropy loss and CIDEr optimization. In order to fairly compare and convincingly prove the effectiveness of our methods, we incorporate our proposed methods into Up-Down model (Anderson et al. 2018) and take it as our baseline model. It is shown that our approaches have obtained remarkable improvement over our baseline model. Our methods achieve BLEU-4 / METEOR / ROUGE-L / CIDEr / SPICE scores of 36.6 / 28.0 / 56.9 / 116.9 / 21.3 with cross-entropy loss and 38.5 / 28.7 / 58.4 / 129.1 / 22.4 with CIDEr optimization. We also conducted a series of experiments, taking several state-of-the-art models as baseline model and introducing our proposed methods into them respectively, which confirmed the effectiveness and generality.

Related Work

Image captioning. Most modern computer vision methods (Socher et al. 2014; Karpathy and Fei-Fei 2015) encode image through CNNs and then decode it with RNNs. (Vinyals et al. 2015) firstly incorporated attention mechanism into captioning. In this way, decoder can better extract local information from image, thus visual features can be better represented. Adaptive attention (Lu et al. 2017) introduced a sentinel gate mechanism into visual attention, which can prompt the extent model focuses on visual features or semantic context. (Yao et al. 2017) added the attributes information to the captioning model, which has greatly improved the performance of object description in captioning. SCA-CNN (Chen et al. 2017) has introduced channel-wise attention into captioning model. By this way, visual features can be better gathered by focusing on crucial channels. (Anderson et al. 2018) have employed Faster R-CNN network pre-trained on Visual Genome (Krishna et al. 2017) to generate more explicit features. Several region features with high confidence gathered as visual feature, which has shown remarkable advantage over CNN feature. (Rennie et al. 2017) have applied reinforcement learning method to captioning. By this way, the model can be optimized directly on those objective evaluation metrics like CIDEr (Vedantam, Lawrence Zitnick, and Parikh 2015), BLEU (Papineni et al. 2002) and etc. The consistency between training objective and evaluation metric has improved caption performance on evaluation scores. (Luo et al. 2018) have incorporated a discriminative loss in reinforcement learning step, which has enhanced the diversity of caption.

Image-text matching. Image-text matching is used to evaluate the relevance between image and text, so we employ image-text matching model to accomplish our text-retrieval task. There have been numerous studies exploring encoding whole image and full sentences to common semantic space for image-text matching. Learning cross-view representations with a hinge-based triplet ranking loss was first attempted by (Kiros, Salakhutdinov, and Zemel 2014a). Images and sentences are encoded by deep Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) respectively. (Faghri et al. 2017) addressed hard negative cases in the triplet loss function and achieve notable improvement. (Gu et al. 2018a) proposed a method integrating generative objectives with the cross-view feature embedding learning. (Lee et al. 2018) suggested the alignment between objects or other stuffs in images and the corresponding words in sentences.

Pointing mechanism. Inspired by pointing mechanism (See, Liu, and Manning 2017), an adaptive and soft switch is applied in this paper to accommodate the word probabilities between generate mode (SG) and copy mode (RWS). In this way, captioning model can switch freely between SG and RWS.

Methods

In this section, we will present our proposed methods in detail. For gaining better performance of caption, as in (Anderson et al. 2018), we extract the R-CNN feature for image: $V = \{v_1, v_2, ..., v_k\}, v_i \in \mathbb{R}^D$. We take the mean pooling vector $\bar{v}$ as the global visual feature.

Text-Retrieval module

For a given image I, the mean pooling vector $\bar{v}$ is taken as the visual feature. We embed it by a fully connected layer W_I:

$$f(I) = W_I\bar{v} \tag{1}$$

For a caption C, we utilize word2vec (Mikolov et al. 2013) to embed words in C:

$$c_i = \mathrm{word2vec}(w_i) \tag{2}$$

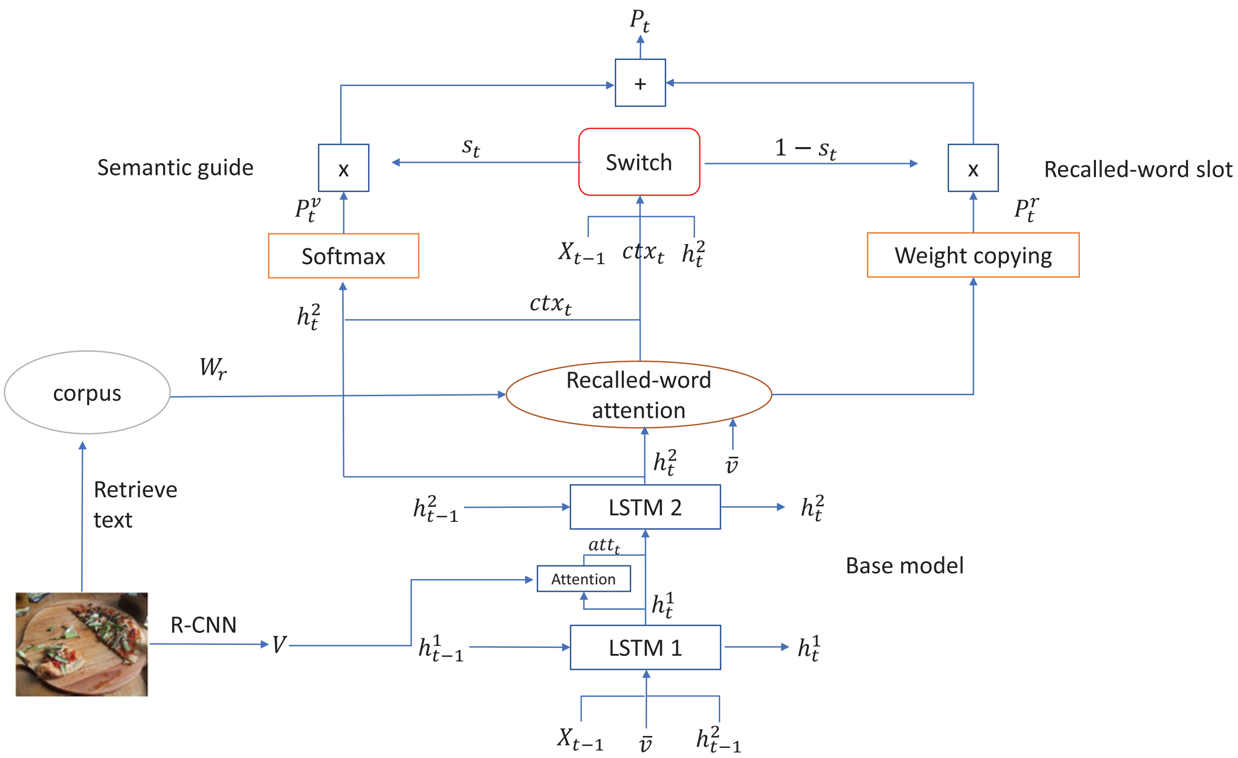


Figure 2: An illustration of our proposed recall mechanism. There are 3 parts in this figure: base model, semantic guide (SG), and recalled-word slot (RWS). We incorporate our recall mechanism into the base model Up-Down (Anderson et al. 2018). SG and RWS compute word probability individually, then we employ a soft switch to combine them into a final word probability P_t.

Where c_i is the embedding of word w_i, we denote $\hat{C} = \{c_1, ..., c_n\}$, where n is the number of words in the caption. Then we feed those words embedding into RNN:

$$S = \text{RNN}(\hat{C}) \tag{3}$$

The output vectors of RNN are $S = \{s_1, ..., s_n\}$, where s_i represents i-th word feature through RNN. Then we apply attention mechanism to predict the weights of word features with respect to image feature $f(I)$. For each vector s_i in S, the weight is calculated as following:

$$u_{i,t} = w_s tanh(W_{s,u} s_i + W_v \bar{v}) \tag{4}$$

$$\alpha_t = softmax(u_t) \tag{5}$$

Where w_s , $W_{s,u}$ and W_v are learned parameters for attention part, and α_t are the weights of word features. Then we take the weighted sum of word features as the feature for caption C:

$$g(C) = \sum_{i=1}^{n} \alpha_{i,t} s_i \tag{6}$$

The similarity score between image I and caption C is computed as the cosine similarity:

$$s(I, C) = \frac{f(I) \cdot g(C)}{\|f(I)\| \, \|g(C)\|} \tag{7}$$

The triplet loss is the ranking objective for our text-retrieval model, we employ a hard negative hinge-based triplet loss with margin α as in (Lee et al. 2018):

$$\begin{aligned} L_{tri}(C, I) = & \max_{\hat{C}}[\alpha + s(I, \hat{C}) - s(I, C)]_+ \\ & + \max_{\hat{I}}[\alpha + s(\hat{I}, C) - s(I, C)]_+ \end{aligned} \tag{8}$$

Where $[x]_+ \equiv max(x, 0)$. For a mini-batch, pairs (I, C) are positive pairs, and pairs $(\hat{I}, C)$ and pairs $(I, \hat{C})$ are negative pairs. The hard negative triplet loss in Eq.(8) tries to make positive pairs (I, C) score higher than the maximum score of negative pairs with a margin α.

Captioning model

As shown in Figure 2, there are 3 parts of our captioning model: base model, semantic guide, and recalled-word slot.

Base model We take Up-Down model (Anderson et al. 2018) as our base model since its remarkable performance. Our proposed recall mechanism is implemented on it. So we will briefly introduce Up-Down model. As illustrated in Figure 2, for each step, the input of LSTM1 is a concatenated vector of last word embedding X_{t-1}, the global pooling vector $\bar{v}$, and the last output h_{t-1}^2 from LSTM2. We attend the hidden output vector h_t^1 of LSTM1 to visual feature V, then feed the concatenated vector of visual context att_t and h_t^1

into LSTM2. We denote [.] as the concatenation operation. The detailed formulas are explained as following:

$$h_t^1 = \mathrm{LSTM1}([X_{t-1}, \bar{v}, h_{t-1}^2]) \tag{9}$$

$$g_{i,t} = w_v tanh(W_v^1 v_i + W_h^1 h_t^1) \tag{10}$$

$$\alpha_t^v = softmax(g_t) \tag{11}$$

$$att_t = \sum_{i=1}^{k} \alpha_{i,t}^v v_i \tag{12}$$

$$h_t^2 = \mathrm{LSTM2}([att_t, h_t^1]) \tag{13}$$

Where W_v^1, W_h^1 and w_v are learned parameters for visual attention part. $\alpha_t^v = \{\alpha_{1,t}, \alpha_{2,t}, ..., \alpha_{k,t}\} \in \mathbb{R}^k$ are the relevant weights of V, which sum to 1. att_t is the weighted sum of V, which represents the most relevant locations of words to generate.

Semantic guide and recalled-word slot By applying our text-retrieval module to corpus, certain recalled words are collected for each image: $W_r = \{w_{r,1}, w_{r,2}, ..w_{r,m}\}$, m is the number of recalled words for an image. $X_r = \{x_{r,1}, x_{r,2}, ..x_{r,m}\}$ is the corresponding embedding for recalled words.

As illustrated in Figure 2, firstly, we apply attention mechanism to obtain the weights of recalled words. The reason why we choose the concatenated vector $[h_t^2, \bar{v}]$ to attend X_r is we regard $[h_t^2, \bar{v}]$ as an unit of the semantic and visual information, which can be an accurate guide. The weights of recalled words are calculated as following:

$$r_{i,t} = w_x tanh(W_r x_i + W_h^2 h_t^2 + W_v^2 \bar{v}) \tag{14}$$

$$\alpha_t^r = softmax(r_t) \tag{15}$$

Where W_r, W_v^2, W_h^2 and w_x are learned parameters in recalled-word attention module, $\alpha_t^r = \{\alpha_{1,t}, \alpha_{2,t}, ..., \alpha_{m,t}\} \in \mathbb{R}^m$ represents the weights in recalled words.

We then apply the weights α_t^r into two branches: semantic guide, and recalled-word slot. In the semantic guide branch, we obtain the recalled content ctx_t by the weighted sum of recalled-word embedding, which can help to generate word probability distribution P_t^v:

$$ctx_t = \sum_{i=1}^{m} \alpha_{i,t}^r x_i \tag{16}$$

$$P_t^v = softmax(W_l[ctx_t, h_t^2]) \tag{17}$$

Where W_l is a logit layer to predict word probability $P_t^v(w)$ by softmax function.

In recalled-word slot, we introduce a weight copying layer to copy the weights from recalled words directly to the word probability distribution P_t^r:

$$P_t^r(w) = \begin{cases} \alpha_t^r(w) & w \in W_r \\ 0 & w \notin W_r \end{cases} \tag{18}$$

Where $\alpha_t^r(w)$ is the weight of word w in W_r. If word w does not exist in W_r, we set 0 to its probability. $P_t^r(w)$ represents the word probability from RWS branch. Since $P_t^r(w)$ only keeps the probability of recalled words, it seems to build a slot from recalled words to output caption.

We compute the final word probability by integrating two probabilities by a soft switch:

$$\begin{aligned} s_t =& \sigma(W_{s,h} h_t^2 + W_{s,c} ctx_t + \\ & W_{s,x} X_{t-1} + b_s) \end{aligned} \tag{19}$$

$$P_t(w) = (1 - s_t) P_t^v(w) + s_t P_t^r(w) \tag{20}$$

Where $\sigma(.)$ is a sigmoid function. $W_{s,h}, W_{s,c}, W_{s,x}$ and bias b_s are the learned parameters to compute $s_t \in [0, 1]$, which is considered as a soft switch between semantic guide and recalled-word slot. $P_t(w)$ is the weighted-sum probability of $P_t^v(w)$ and $P_t^r(w)$.

Take a overall look at our proposed methods. In semantic guide branch, ctx_t is considered as a recalled content, merged with hidden vector h_t^2 to generate words. In recalled-word slot, words generation are conducted by copying weights directly. Therefore, the semantic guide branch generates word by "deep consideration" with visual and recalled information, and the recalled-word slot is more like "intuition". The soft switch s_t plays a role in combining them together to choose the most reasonable words.

Objective

Given an image I, a target ground truth sequence $w_{1:T}^*$. and an image captioning model with parameters θ, we minimize the cross entropy loss as following:

$$\begin{aligned} L_{mle}(\theta) &= -\sum_{t=1}^{T} \log(p_\theta(w_t^* | w_{1:t-1}^*)) \\ &= -\sum_{t=1}^{T} \log(P_t(w_t^* | w_{1:t-1}^*)) \\ &= -\sum_{t=1}^{T} \log\big(\,(1 - s_t) P_t^v(w_t^* | w_{1:t-1}^*) \\ &\quad + s_t P_t^r(w_t^* | w_{1:t-1}^*)\,\big) \end{aligned} \tag{21}$$

Where w_t^* is the word from the ground truth sequence at step t. In order to boost performance of captioning model, and to compare with recent work (Anderson et al. 2018; Rennie et al. 2017), we also apply CIDEr optimization to our training process. Initializing from the cross entropy model, the traditional CIDEr optimization (Rennie et al. 2017) approach focuses on optimizing the CIDEr scores of the generated sentences. The training process is to minimize negative expected reward:

$$L_r(\theta) = -\mathbb{E}_{w_{1:T} \sim p_\theta}[r(w_{1:T})] \tag{22}$$

Where r is the CIDEr function that can give a score for a sequence $w_{1:T}$. Following the method in SCST (Rennie et al. 2017), we approximate the gradient as:

$$\begin{aligned} \nabla_\theta L_r(\theta) \approx & -(r(w_{1:T}^s) - r(w_{1:T}^g)) \times \\ & \nabla_\theta \log(p_\theta(w_{1:T}^s)) \end{aligned} \tag{23}$$

Where $w_{1:T}^s$ is the sampled caption from the final probability distribution $P_t(w)$, and $w_{1:T}^g$ represents the caption obtained by greedily decoding. We take the score $r(w^{g}_{1:T})$ as the baseline, which is used to reduce the variance in training process.

Recalled-word reward Due to the fact that there are two branch probabilities in our model, we make a change to boost traditional CIDEr optimization. We propose an individual reward for recalled-word slot:

$$r(W_r) = r(w^{s}_{1:T}) - r(w^{\hat{s}}_{1:T}) \tag{24}$$

Where $w^{s}_{1:T}$ is a sampled caption from the final probability distribution $P_t(w)$, and $w^{\hat{s}}_{1:T}$ represents the sampled caption with soft switch $s_t = 0$ at all steps in generation. $s_t = 0$ indicates that we cut off recalled-word slot and only sample caption from $P^{v}_{t}(w)$. In this way, recalled-word reward $r(W_r)$ can certify how much the improvement is from recalled-word slot. We minimize negative expected reward as following:

$$\begin{aligned} L_r(\theta) = &-\lambda \mathbb{E}_{w^{\hat{s}}_{1:T} \sim p^{v}_{\theta}}[r(w^{\hat{s}}_{1:T})] \\ &-(1-\lambda)\mathbb{E}_{w^{s}_{1:T} \sim p_{\theta}}[r(W_r)] \end{aligned} \tag{25}$$

$$\begin{aligned} \nabla_\theta L_r(\theta) \approx &-\lambda(r(w^{\hat{s}}_{1:T}) - r(w^{\hat{g}}_{1:T}))\times \\ &\nabla_\theta \log(p^{v}_{\theta}(w^{\hat{s}}_{1:T})) - \\ &(1-\lambda) r(W_r) \nabla_\theta \log(p_\theta(w^{s}_{1:T})) \end{aligned} \tag{26}$$

Where $w^{\hat{g}}_{1:T}$ represents the caption obtained by greedily decoding by cutting off recalled-word slot, which is viewed as the baseline for $w^{\hat{s}}_{1:T}$. We do not introduce a baseline for $r(W_r)$, because it is originally the result of subtraction, so the variance is relatively small.

Experiments and results

Datasets

MSCOCO We use the MSCOCO 2014 captions dataset (Lin et al. 2014) to evaluate our proposed method. As the largest English image caption dataset, MSCOCO contains 164,062 images. In this paper, we employ the 'Karpathy' splits (Karpathy and Fei-Fei 2015) for validation of model hyperparameters and offline evaluation. This split has been widely used in prior works, choosing 113,287 images with five captions each for training and 5000 respectively for validation and test. For quantitative performance evaluation, we use the standard automatic evaluation metrics, namely SPICE (Anderson et al. 2016), CIDEr (Vedantam, Lawrence Zitnick, and Parikh 2015), METEOR (Denkowski and Lavie 2014), ROUGE-L (Lin 2004) and BLEU (Papineni et al. 2002)

Visual Genome Visual Genome (VG) (Goyal et al. 2017) datasets contains 5.4M region descriptions for 108K images and 42 for each image on average. Each description phrase varies from 1 to 16 words. The dataset is densely annotated with scene graphs containing bounding boxes, classifications and attributes of main objects, and the relationships among different instances. Totally, it contains 3.8 million object instances, 2.8 million attributes and 23 million relationships. The R-CNN feature pre-trained (Anderson et al. 2018) on VG dataset is employed in our experiment.

Implementation Details

For a more fair and convincing comparison, we use the R-CNN feature in Up-Down (Anderson et al. 2018) as our image feature. For each image, 10 to 100 ROI (region of interest) pooling vectors are preserved. Each vector size is 2048. We use this R-CNN feature both in text-retrieval module and captioning model. Following previous methods in captioning, we do not use pretrained word embeddings, and all word embeddings are trained from scratch.

In our text-retrieval module, we adopt the mean pooling vector of R-CNN feature, and project it to a hidden vector of size 1024. Then we apply Bi-LSTM with size 512 to embed text. The word embedding size is also 512, so the output vector of Bi-LSTM is 1024, which matchs the hidden vector size of image. During training, the batch size is set to 128, and the margin α is set to 0.2. The learning rate is set to 5e-4 and decay by a factor 0.8 for every 3 epochs.

In captioning model, our base model is Up-Down model, so we use the same hyper-parameters as Up-Down model. The hidden units of LSTM1 and LSTM2 are both 1024, and the size of word embedding is also 1024. We adopt Adam optimizer with the learning rate set as 5e-4 and decay also by a factor 0.8 for every 3 epochs, and the batch size is set to 64. For CIDEr optimization training, we initialize the learning rate as 5e-5, decaying by a factor 0.1 for every 50 epochs. We choose the best model from cross-entropy training for the following CIDEr optimization training. GeForce GTX 1080Ti is the GPU we employed in all experiments.

Performance of text-retrieval model

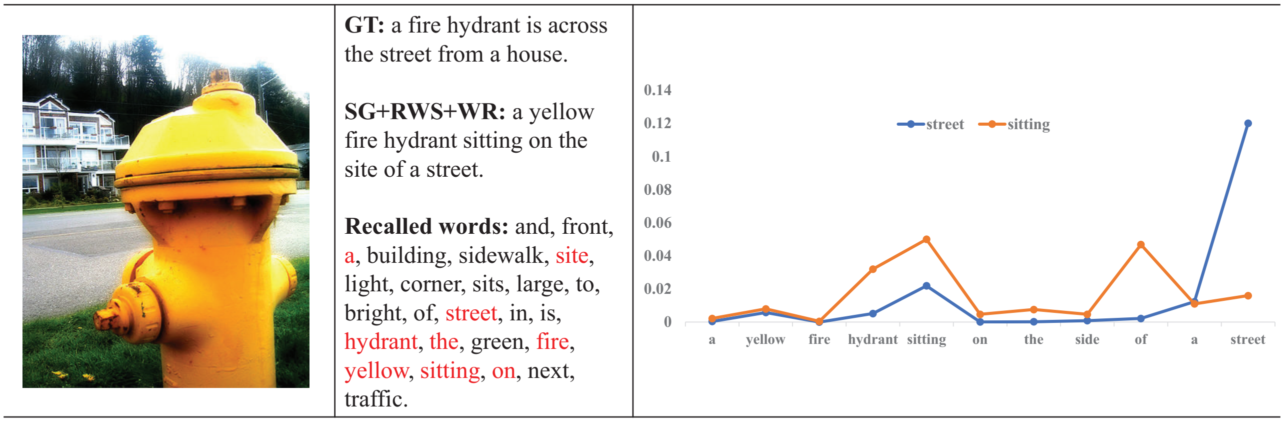
The performance of text-retrieval model can directly affect the relevance between the recalled words and image. So we evaluate its performance on the validation set of MSCOCO Karpathy splits, and the result is reported in Table 3. We assume that our text-retrieval model does not achieve state-of-the-art as in (Lee et al. 2018), but it is well qualified to retrieve sufficient and relevant words for images. Our corpus is collected from all the captions of MSCOCO Karpathy train splits. For determining an appropriate number of captions to be retrieved, we respectively retrieve top 1, 5 and 15 related captions for each image, and test the performance on cross-entropy loss. Table 4 shows that top 5 captions retrieved for each image is a better choice than 1 or 15. It is necessary to emphasize that we avoid retrieving the ground truth caption for images, and all the retrieved captions only come from the train splits of MSCOCO Karpathy. In the following experiments, top 5 captions retrieved are used to construct recalled words for each image.

Captioning performance

Selection of λ λ in Eq.(25) is the trade-off parameter in CIDEr optimization training, which balances the reward between $r(w^{\hat{s}}_{1:T})$ and $r(W_r)$ in loss function. Thus we set different λ values from 0 to 1 to conduct a model selection. Experiments for λ are based on the best performance model (K=5) in Table 4. The experiment result is reported in Table 5, which shows that $\lambda = 0.5$ has the best performance out of others. As a result, in the following experiments, we set $\lambda = 0.5$.

	Cross-entropy loss						CIDEr optimization training					
Models	B-1	B-4	M	R	C	S	B-1	B-4	M	R	C	S
Test-guide (Mun, Cho, and Han 2016)	74.9	32.6	25.7	-	102.4	-	-	-	-	-	-	-
SCST (Rennie et al. 2017)	-	30.0	25.9	53.4	99.4	-	-	34.2	26.7	55.7	114.0	-
StackCap (Gu et al. 2018a)	76.2	35.2	26.5	-	109.1	-	78.5	36.1	27.4	-	120.4	-
CAVP (Liu et al. 2018)	-	-	-	-	-	-	-	**38.6**	28.3	**58.5**	126.3	21.6
Up-Down (Anderson et al. 2018)	**77.2**	36.2	27.0	56.4	113.5	20.3	79.8	36.3	27.7	56.9	120.1	21.4
Ours:SG	77.1	36.3	27.8	56.8	115.3	21.0	80.2	38.3	28.5	58.3	127.3	22.0
Ours:SG+RWS	77.1	36.6	28.0	56.9	116.9	21.3	80.3	38.3	28.5	58.3	128.3	22.2
Ours:SG+RWS+WR	77.1	**36.6**	**28.0**	**56.9**	**116.9**	**21.3**	**80.3**	38.5	**28.7**	58.4	**129.1**	**22.4**

Table 1: Experiment of our proposed recall mechanism on the MSCOCO Karpathy test split with both cross-entropy loss and CIDEr optimization. We implement our proposed methods: semantic guide (SG), recalled-word slot (RWS) and recalled-word reward (WR) on the baseline model Up-Down. Test results show that our proposed methods have obvious improvement over our baseline. B-1 / B-4 / M / R / C / S refers to BLEU1/ BLEU4 / METEOR / ROUGE-L / CIDEr / SPICE scores.



GT: a fire hydrant is across the street from a house.

SG+RWS+WR: a yellow fire hydrant sitting on the site of a street.

Recalled words: and, front, a, building, sidewalk, site, light, corner, sits, large, to, bright, of, street, in, is, hydrant, the, green, fire, yellow, sitting, on, next, traffic.

Figure 3: The left part: recalled words and caption generated by SG+RWS+WR. The right part: visualization of the weights in recalled-word attention

Models	B-4	M	R	C	S
att2in (Rennie et al. 2017)	36.1	27.2	56.9	119.1	20.8
att2in+SG+RWS+WR	**36.7**	**27.8**	**57.4**	**122.0**	**21.4**
att2all (Rennie et al. 2017)	36.3	27.5	57.2	121.7	21.1
att2all+SG+RWS+WR	**37.1**	**28.0**	**57.8**	**125.0**	**21.7**
stackcap (Gu et al. 2018a)	36.6	27.6	57.3	121.1	21.0
stackcap+SG+RWS+WR	**37.8**	**28.3**	**58.0**	**126.4**	**21.9**

Table 2: Performance of our proposed methods over other state-of-the-art models after cider optimization training.

	R@1	R@5	R@10	Mean r
Text-retrieval	36.2	69.0	81.7	7.7
Image-retrieval	39.6	72.3	83.7	11.0

Table 3: Performance of text-retrieval model on MSCOCO Karpathy validation set.

	Cross-Entropy Loss					
Top K	B-1	B-4	M	R	C	S
K=1	77.1	36.3	27.8	56.9	115.8	21.2
K=5	77.1	36.5	28.0	57.0	**116.7**	21.3
K=15	77.0	36.3	27.9	56.6	115.6	21.0

Table 4: Experiments on choice of K, the number of captions retrieved for each image. B-1 / B-4 / M / R / C / S refers to BLEU1/ BLEU4 / METEOR / ROUGE-L / CIDEr / SPICE scores. Experiments are conducted on MSCOCO Karpathy validation set.

Evaluation of proposed methods From above, we have proposed semantic guide (SG), recalled-word slot (RWS) in captioning model and recalled-word reward (WR) in CIDEr optimization. Then we test their performances on MSCOCO Karpathy test split. Beam search with beam size 2 is employed to generate captions. The performances of proposed methods are shown in Table 1, and our baseline is Up-Down model in (Anderson et al. 2018). Focusing on the CIDEr score at Table 1, semantic guide (SG) has improved 1.5% with cross-entropy loss, and 6.0% on CIDER optimization. Semantic guide with recalled-word slot (SG+RWS) has improved 3.0% on cross-entropy loss, and 6.8% on CIDEr optimization. Our best model (SG+RWS+WR) has obtained 7.5% improvement on CIDEr optimization, and it has obtained BLEU4 / CIDEr / SPICE scores of 36.6 / 116.9 / 21.3 with cross-entropy loss and 38.5 / 129.1/ 22.4 with CIDEr optimization. In addition, compared with other state-of-the-art models, like SCST (Rennie et al. 2017), StacKCap (Gu et al. 2018b), and CAVP (Liu et al. 2018), our results still out-

	Cider optimization training					
λ	B-1	B-4	M	R	C	S
0.1	80.2	38.2	28.5	58.2	127.5	22.3
0.3	80.2	38.4	28.6	58.1	128.1	22.3
0.5	80.3	38.4	28.6	58.3	**128.9**	22.4
0.7	80.3	38.1	28.6	58.2	127.8	22.2
0.9	80.1	38.0	28.4	58.1	127.3	22.2

Table 5: Experiments on choice of λ, the trade-off parameter in CIDEr optimization. B-1 / B-4 / M / R / C / S refers to BLEU1/ BLEU4 / METEOR / ROUGE-L / CIDEr / SPICE scores. Experiments are conducted on MSCOCO Karpathy validation set

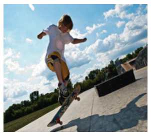
Image	Recalled word	Caption
	plane airport blue small white etc.	**GT**: a small blue plane sitting on top of a field. **Up-Down**: a small plane is sitting on the grass. **SG+RWS+WR**: a small blue and white plane sitting in the grass.
	building front in of buildings bus etc.	**GT**: a group of people in front of a white building. **Up-Down**: a group of people standing in a street with a clock. **SG+RWS+WR**: a group of people standing in front of a building with a clock.
	trick ramp boy jump riding park etc.	**GT**: a young boy is performing tricks on a skateboard. **Up-Down**: a young man riding a skateboard on a ramp. **SG+RWS+WR**: a young boy doing a trick on a skateboard on a ramp.

Figure 4: Recalled word and caption generation results on MS COCO Karpathy test split and the output sentences are generated by 1) Ground Truth (GT): one ground truth caption, 2)Up-Down model and 3) our SG+RWS+WR model.

perform theirs. The results of comparison demonstrate the effectiveness of our proposed methods, especially on those more convincing evaluation metrics such as CIDEr, SPICE.

To prove the effectiveness and generality of our proposed methods, we have also implemented our proposed methods over other state-of-the-art models: att2in(Rennie et al. 2017), att2all(Rennie et al. 2017) and stackcap (Gu et al. 2018a). We do the comparative experiments over these three models. As is shown in Table 2, the results indicate that our proposed methods have a wide range of applicability to many state-of-art models. To be detailed, we have average 2.3% improvement on att2in, 2.1% improvement on att2all, and 3.3% improvement on stackcap. We have conducted the MSCOCO online evaluation and achieved promising results (called "caption-recall", reported at 19 Oct 2019), which also surpass the online results of our baseline model Up-Down.

Qualitative Analysis To help qualitatively evaluate the effectiveness of our recall mechanism, Figure 3 and Figure 4 show some examples generated by our recall mechanism. As shown in Figure 4, we can observe that recalled words whose font in red color, like: "blue", "front", "building" and "boy", are generated in captions by SG+RWS+WR, and those words are also in ground truth, but not in the caption generated by the base model Up-Down. This illustrates that our recall mechanism make the generated sentence closer to ground truth caption. Moreover, there are some recalled words in blue color that are not in ground truth caption, but they are highly consistent with the image. In Figure 3, we present all the recalled words for an image. In this example, each word in generated caption can be found in recalled words. This proves the high correlation between generated caption and recalled words. Recalled-word slot and semantic guide are highly dependant on the weights in recalled-word attention, so we also visualize the weights of "sitting" and "street" at each generation step. We can observe that two words weight attain the max when they are generated.

Conclusion

In this paper, we introduce a novel recall mechanism to the captioning model. In our recall mechanism, a recall unit is designed to retrieve recalled words for image, and semantic guide and recalled-word slot are proposed to make full use of recalled words. A recalled-word reward is introduced to boost CIDEr optimization. The experiments prove that our recall mechanism can effectively employ recalled information to improve the quality of generated caption.

It needs to be emphasized that our proposed methods can be applied to any captioning model. Meanwhile, using training data both for model training and retrieving recalled words, can be instructive to other researches of captioning.

Acknowledgement

This paper is supported by NSFC (No.61772330, 61533012, 61876109), the Basic Research Project of Innovation Action Plan (16JC1402800), the advanced research project (No.61403120201), Shanghai authentication key Lab. (2017XCWZK01),Technology Committee the interdisciplinary Program of Shanghai Jiao Tong University (YG2015MS43), and AI-Lab VS Team, Bytedance.

References

Anderson, P.; Fernando, B.; Johnson, M.; and Gould, S. 2016. Spice: Semantic propositional image caption evaluation. In *European Conference on Computer Vision*, 382–398. Springer.

Anderson, P.; He, X.; Buehler, C.; Teney, D.; Johnson, M.; Gould, S.; and Zhang, L. 2018. Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 6077–6086.

Chen, L.; Zhang, H.; Xiao, J.; Nie, L.; Shao, J.; Liu, W.; and Chua, T.-S. 2017. Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 5659–5667.

Denkowski, M., and Lavie, A. 2014. Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, 376–380. Baltimore, Maryland, USA: Association for Computational Linguistics.

Faghri, F.; Fleet, D. J.; Kiros, J. R.; and Fidler, S. 2017. Vse++: Improved visual-semantic embeddings. *arXiv preprint arXiv:1707.05612* 2(7):8.

Goyal, Y.; Khot, T.; Summers-Stay, D.; Batra, D.; and Parikh, D. 2017. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 6904–6913.

Gu, J.; Cai, J.; Joty, S. R.; Niu, L.; and Wang, G. 2018a. Look, imagine and match: Improving textual-visual cross-modal retrieval with generative models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 7181–7189.

Gu, J.; Cai, J.; Wang, G.; and Chen, T. 2018b. Stack-captioning: Coarse-to-fine learning for image captioning. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Karpathy, A., and Fei-Fei, L. 2015. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3128–3137.

Kiros, R.; Salakhutdinov, R.; and Zemel, R. S. 2014a. Unifying visual-semantic embeddings with multimodal neural language models. *arXiv preprint arXiv:1411.2539*.

Kiros, R.; Salakhutdinov, R. R.; and Zemel, R. S. 2014b. Unifying visual-semantic embeddings with multimodal neural language models. *CoRR* abs/1411.2539.

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. *International Journal of Computer Vision* 123(1):32–73.

Lee, K.-H.; Chen, X.; Hua, G.; Hu, H.; and He, X. 2018. Stacked cross attention for image-text matching. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 201–216.

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 *European conference on computer vision*, 740–755. Springer.

Lin, C.-Y. 2004. Rouge: A package for automatic evaluation of summaries. *Text Summarization Branches Out*.

Liu, D.; Zha, Z.-J.; Zhang, H.; Zhang, Y.; and Wu, F. 2018. Context-aware visual policy network for sequence-level image captioning. *arXiv preprint arXiv:1808.05864*.

Lu, J.; Xiong, C.; Parikh, D.; and Socher, R. 2017. Knowing when to look: Adaptive attention via a visual sentinel for image captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 375–383.

Lu, J.; Yang, J.; Batra, D.; and Parikh, D. 2018. Neural baby talk. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 7219–7228.

Luo, R.; Price, B.; Cohen, S.; and Shakhnarovich, G. 2018. Discriminability objective for training descriptive captions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 6964–6974.

Mikolov, T.; Chen, K.; Corrado, G.; and Dean, J. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mun, J.; Cho, M.; and Han, B. 2016. Text-guided attention model for image captioning. In *AAAI*.

Papineni, K.; Roukos, S.; Ward, T.; and Zhu, W.-J. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, 311–318. Association for Computational Linguistics.

Qin, Y.; Du, J.; Zhang, Y.; and Lu, H. 2019. Look back and predict forward in image captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 8367–8375.

Rennie, S. J.; Marcheret, E.; Mroueh, Y.; Ross, J.; and Goel, V. 2017. Self-critical sequence training for image captioning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

See, A.; Liu, P. J.; and Manning, C. D. 2017. Get to the point: Summarization with pointer-generator networks. *arXiv preprint arXiv:1704.04368*.

Socher, R.; Karpathy, A.; Le, Q. V.; Manning, C. D.; and Ng, A. Y. 2014. Grounded compositional semantics for finding and describing images with sentences. *Transactions of the Association for Computational Linguistics* 2:207–218.

Vedantam, R.; Lawrence Zitnick, C.; and Parikh, D. 2015. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4566–4575.

Vinyals, O.; Toshev, A.; Bengio, S.; and Erhan, D. 2015. Show and tell: A neural image caption generator. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3156–3164.

Xu, K.; Ba, J.; Kiros, R.; Cho, K.; Courville, A.; Salakhudinov, R.; Zemel, R.; and Bengio, Y. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, 2048–2057.

Yao, T.; Pan, Y.; Li, Y.; Qiu, Z.; and Mei, T. 2017. Boosting image captioning with attributes. In *Proceedings of the IEEE International Conference on Computer Vision*, 4894–4902.

You, Q.; Jin, H.; Wang, Z.; Fang, C.; and Luo, J. 2016. Image captioning with semantic attention. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.