Relation Also Need Attention: Integrating Relation Information Into Image Captioning

Tianyu Chen TYCHEN@STU.GXNU.EDU.CN
Zhixin Li* LIZX@GXNU.EDU.CN
Tiantao Xian XIANTT@STU.GXNU.EDU.CN
Canlong Zhang CLZHANG@GXNU.EDU.CN
Guangxi Key Lab of Multi-source Information Mining and Security,
Guangxi Normal University, Guilin 541004, China

Huifang Ma MAHUIFANG@NWNU.EDU.CN
College of Computer Science and Engineering,
Northwest Normal University, Lanzhou 730070, China

Editors: Vineeth N Balasubramanian and Ivor Tsang

Abstract

Image captioning methods with attention mechanism are leading this field, especially models with global and local attention. But there are few conventional models to integrate the relationship information between various regions of the image. In this paper, this kind of relationship features are embedded into the fused attention mechanism to explore the internal visual and semantic relations between different object regions. Besides, to alleviate the exposure bias problem and make the training process more efficient, we combine Generative Adversarial Network with Reinforcement Learning and employ the greedy decoding method to generate a dynamic baseline reward for self-critical training. Finally, experiments on MSCOCO datasets show that the model can generate more accurate and vivid image captioning sentences and perform better in multiple prevailing metrics than the previous advanced models.

Keywords: Image Captioning; Fused Attention Mechanism; Generative Adversarial Network; Sequence-level Training; Reinforcement Learning

1. Introduction

Automatic image captioning intends to generate a descriptive sentence that verbalizes the visual content of an image. The current encoder-decoder model based on Convolutional Neural Network (CNN) with attention mechanism and Recurrent Neural Network (RNN) has been leading this field. However, the RNN model faces a common problem in dealing with the sequence generation problem: Exposure Bias, which will influence the result inevitably.

Most traditional global attention mechanisms allocate attention weights only to CNN's low-level coarse features. It may cause the objects in the picture to be mistakenly translated into words. What's more important, the crucial clues of the relationship with important guidance between different objects are also neglected. Concerning the caption generation

* Zhixin Li is the corresponding author(lizx@gxnu.edu.cn).

part, there are certain drawbacks associated with the application of Generative Adversarial Network (GAN) Creswell et al. (2018) in discrete tokens generation. A major reason is that the generative model's discrete outputs make it difficult to pass the gradient update from the discriminative model to the generative model. The solution was then assayed for SeqGAN Yu et al. (2017) model, which combines GAN with policy gradient algorithm pg. Nevertheless, when the policy is already powerful, the model may still sample a bad sentence. The probability of this sentence will even increase because it still has a reward value.

In order to solve the above problems, this paper proposes a Global Local-Relation Attention(GLRA) mechanism to excavate the image's information more effectively. The important relationship features are allocated with attention weights in this model. Besides, the GAN is trained in the way of self-critical to solve the exposure bias problem. The main contributions of this paper are as follow:

- Unlike the previous method, we propose a variant of self-attention Vaswani et al. (2017) to integrating relation information between different image regions into global features.
- The relationship features contain visual similarity and semantic information between different objects are integrated into the local attention mechanism. These three kinds of features complement each other to more fully excavate and represent the feature information of the image.
- In the language part, the RL algorithm is combined with GAN. Simultaneously, the greedy decoding method is applied to optimize the model structure through self-critical training by providing a dynamic baseline reward value.
- The experimental results on the MSCOCO dataset show that either of these methods can enhance the experiment performance. Furthermore, when they are integrated, the improvement is more salient. The experimental results exhibit the effectiveness of our approach quantitatively and qualitatively.

2. Related Work

We mainly introduce the application of neural networks with attention mechanism including the model based on the Transformer Vaswani et al. (2017). Besides, some sequence-level learning methods and transformer-based methods are described.

2.1. Attention Mechanism

Inspired by soft-attention mechanism which can focus on diverse parts of the image when generating different words, You et al. (2016) initiated semantic attention. They abstracted important global semantic information from the image to enhance image information. Later, Wang et al. (2019) proposed a hierarchical attention network, which combines patch, target, and text semantic features to enhance image information. Anderson et al. (2018) believed the salient targets in the image should receive more attention, so he improved the traditional method of evenly distributing attention to each region of the image and added bottom-up attention through Faster R-CNN. Yao et al. (2018) initiated the GCN-LSTM architecture, which novelly integrates both semantic and spatial object relationships into image encoder. Huang et al. Huang et al. (2020) innovatively made use of the internal annotation knowledge to assist the calculation of visual attention, then introduced a new strategy to inject external knowledge extracted from knowledge graph into the encoder-decoder framework to facilitate meaningful captioning. The original self-attention proposed by Vaswani et al. (2017) is regard as a great innovation in both Computer Vision and Natural Language Processing. It transforms the input features into three representation $\boldsymbol{Q}$, $\boldsymbol{K}$, and $\boldsymbol{V}$. The calculation formula is as follows:

$$Attention(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \mathrm{softmax}(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d}})\mathbf{V} \tag{1}$$

This method has the advantage to catch the global long-distance relation information and compute parallelly. Wei et al. (2020) combined sentence-level attention with word-level attention for obtaining more detail and accurate captions. Huang et al. (2019) firstly considered whether or how well the attended vector and the given attention query are related, and proposed an "Attention on Attention"(AoA) module which extends the conventional attention mechanisms to determine the relevance between attention results and queries. Liu et al. (2020) proposed an Interactive Dual Generative Adversarial Network(IDGAN), which mutually combined the retrieval-based and generation-based methods to learn a better image captioning ensemble. The experiment results showed the great effectiveness of this model. Zhou et al. (2020) conducted Part-of-Speech enhanced image-text matching model named POS-SCAN, as the effective knowledge distillation for more grounded image captioning. Wang et al. (2020b) introduced the recall mechanism to integrate the prior knowledge of the similar image captions, they first used the text retrieval model to calculate the similarity between the image and other captions in the training set, and the words in the first five captions are selected as recall words to guide the sentence generation. Cornia et al. Cornia et al. (2020) improved the transformer-based model in both image encoding and language generation steps. The proposed meshed-memory transformer can learn a multi-level representation of the relationships between image regions integrating learned prior knowledge. Another excellent work based on Transformer is the X-Linear Attention network Pan et al. (2020) proposed by Pan et al. It initiated a unified X-Linear attention block, which can fully employs bilinear pooling to selectively capitalize on visual information or perform multi-modal reasoning.

2.2. Sequence-level Training

With the aim to solve the exposure bias problem caused by the traditional RNN based decoder, Ranzato et al. (2015) introduced policy gradient algorithm into RNN based sequence generation model for the first time and used Reinforcement Learning combined with the Monte Carlo sampling method for training. Although evaluating the generated result on the sentence-level can alleviate the exposure bias problem to a certain extent, their performance on metric with recall is still unsatisfactory, Chen and Jin (2020) proposed the SLL-SLE and added a sequence-level exploration term to the conventional loss function to boost recall. It guides the model to explore more plausible captions in the training phase. By this means, the proposed sequence-level learning objective takes both the precision and recall sides of generated captions into account. Rennie et al. (2017) proposed a self-critical sequence training method, which employs the sentences generated by the current model as the baseline to reduce the variance of gradient estimation. By this way the model can generate better description sentences than the auxiliary sentences. Yu et al. Yu et al. (2017) innovatively changed the output passed by the discriminator to the generator into a continuous probability value, which presents the probability that generated sentence is ground truth. Referring to the idea of self-critical sequence training(SCST) Rennie et al. (2017), we propose SC-GAN and provide a baseline reward generated by the greedy decoding method, which can not only reduce the high variance of the reward obtained by Monte-Carlo search but also optimize the reward and punishment of each generated sample more clear.

3. Method

Given an raw image, image captioning aims to generate a text description $Y = \{Y_1, Y_2, ...Y_T\}$, where T is the length of sentence. As depicted in Fig. 1, our model consists of the Global Local-Relation Attention(GLRA) and the Self-critical Generative Adversarial Network(SC-GAN). We detail these parts in subsection.

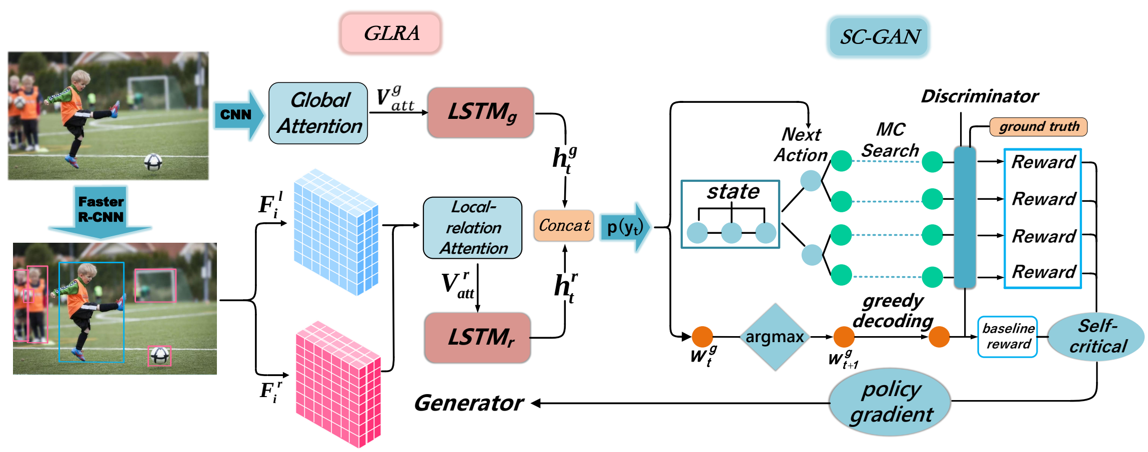

Figure 1: The overview of our proposed system Global Local-Relation Attention(GLRA) and Self-critical GAN(SC-GAN). The GLRA is composed of global attention and local-relation attention. After deriving the next word probability $p(y_t)$ from the GLRA to the SC-GAN, the Discriminator of SC-GAN completes the generated sentence and updates the parameter of Generator by policy gradient strategy.

3.1. Global Local-Relation Attention(GLRA)

Directly processing by CNN is the conventional method for extracting the global features in traditional attention mechanism. But in GLRA, a variant of self-attention is adopted to obtain further in-depth information of global static features. We replace the original formula in (1) for computing the similarity coefficients of the vector $\boldsymbol{Q}$ and the vector $\boldsymbol{K}$

with a single neural network. As shown in Fig. 2. Firstly, the input image is encoded into a spatial feature vector $\boldsymbol{I}$=$(i_1, i_2, ..., i_L)$ by CNN, where L is the number of image space regions. $i_{1:L}\in \mathbb{R}^C$ represents the feature of regions and L=$n\times n$. Afterward three 1×1 convolutional layers $\boldsymbol{W}_q$, $\boldsymbol{W}_k$, and $\boldsymbol{W}_v$ are used to transform $\boldsymbol{I}$ into three spatial features $\boldsymbol{Q}$, $\boldsymbol{K}$, and $\boldsymbol{V}$. Then the attention weights $\boldsymbol{a}$ on $\boldsymbol{V}$ is calculated by fusing $\boldsymbol{Q}$ and $\boldsymbol{K}$. The final global feature $\boldsymbol{V}^g_{att}$ is obtained by multiplying the attention weights $\boldsymbol{a}$ by $\boldsymbol{V}$. The global attention mechanism can expressed by the following formula:

$$\begin{aligned}
&\boldsymbol{Q} = \boldsymbol{W}_q\boldsymbol{I}, \boldsymbol{K} = \boldsymbol{W}_k\boldsymbol{I}, \boldsymbol{V} = \boldsymbol{W}_v\boldsymbol{I} \\
&\boldsymbol{a} = f(\boldsymbol{Q}, \boldsymbol{K}) = \boldsymbol{W}_s(relu(\boldsymbol{Q} + \boldsymbol{K})) + b_s \\
&\boldsymbol{a} = softmax(\boldsymbol{a}^T) \\
&\boldsymbol{V}^g_{att} = \boldsymbol{V}*\boldsymbol{a}
\end{aligned} \tag{2}$$

Where $\boldsymbol{W}_q\in \mathbb{R}^{C'\times C}$, $\boldsymbol{W}_k\in \mathbb{R}^{C'\times C}$ and $\boldsymbol{W}_v\in \mathbb{R}^{C''\times C}$. $\boldsymbol{W}_s\in \mathbb{R}^{C'}$ is the transformation matrix to fuse $\boldsymbol{Q}$ and $\boldsymbol{K}$. And $\boldsymbol{a}, \boldsymbol{V}$ have the same space size, that is, $n\times\ n$. The obtained $\boldsymbol{V}^g_{att}$ which represent the global features with regions' relation information is passed to the global Long short-term memory(LSTM) network, that is LSTM$_g$ in GLRA. Then the corresponding LSTM hidden state h^g_t is generated.

The objects' relation information also play a key role. In our GLRA, Faster R-CNN did the synthesis of local features and relation features between different objects. The detected object regions are top-N Region of Interest(RoI) and expressed as $R_{1:N}$. For object R_i, the local feature is represented as $\boldsymbol{F}^l_i$, which obtained directly by Faster R-CNN. The relation feature is represented as $\boldsymbol{F}^r_i$, which is obtained by multiplying other objects' features with their corresponding weights. We mainly represent the objects' relationship as visual similarity and semantic information. The visual similarity can be calculated by fusing the local features. It is acknowledged that the adjacent object regions usually contain important semantic information, such as "football" and "doorframe", "people" and "football" in Fig. 3, each pair of them should play a key role when generating the other one. These adjacent objects' features will be packed together to distribute attention weights. For object R_i, we select top-K neighbouring objects $R_{1:K}$ according to the IoU(Interaction of Union) and the relative distance between objects. As shown in Fig. 3, the coefficient of R_i and R_j is calculated by dot-product and softmax normalization:

$$f(\boldsymbol{F}^l_i, \boldsymbol{F}^l_j) = \frac{exp(\boldsymbol{F}^l_i \odot \boldsymbol{F}^l_j)}{\sum_{j=1}^{K} exp(\boldsymbol{F}^l_i \odot \boldsymbol{F}^l_j)} \tag{3}$$

$\odot$ represents dot-product operation, the value is further processed by softmax. In this way, the visual similarity between different objects can be excavate. The semantic information is also utilized by the operation of combining the adjacent objects' features. The relation feature of R_j is obtained by:

$$\boldsymbol{F}^r_i = \sum_{j=1}^{K} f(\boldsymbol{F}^l_i, \boldsymbol{F}^l_j)\boldsymbol{F}^l_j \tag{4}$$

F^r_i represent the synthesis of adjacent objects' features. For the semantic relation information between objects with long distance, the above global attention can effectively represent

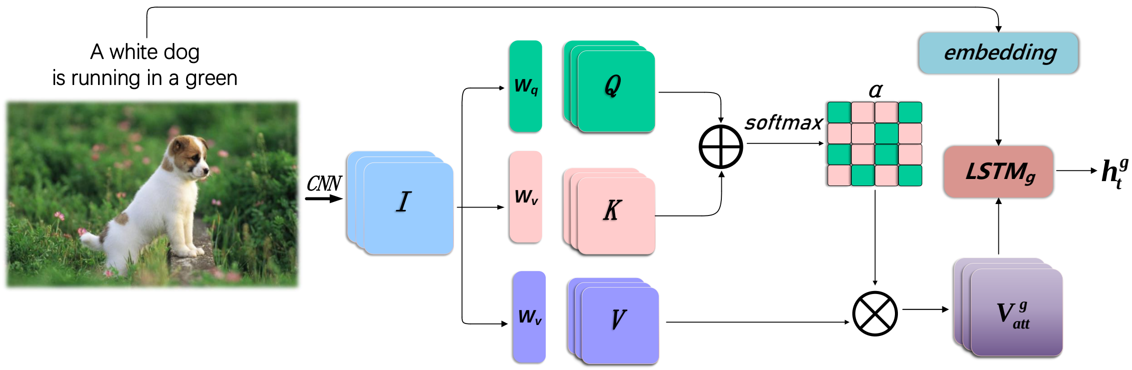

Figure 2: The illustration of the global attention mechanism in GLRA. $\boldsymbol{W}_q$, $\boldsymbol{W}_k$, and $\boldsymbol{W}_v$ are three different 1×1 convolution layers. $\bigoplus$ means fusing the $\boldsymbol{Q}$ and $\boldsymbol{K}$ of different regions by a single neural network. The $\boldsymbol{V}^g_{att}$ is the $\boldsymbol{V}$ assigned with attention weights. $\bigotimes$ means element-wise multiplication. we deliver the embedded previous word and $\boldsymbol{V}^g_{att}$ into the global LSTM to generate the global hidden state $\boldsymbol{h}^g_t$.

them. So far, the model has extracted the local feature and relation feature of each object. The features integrated into the LSTM$_r$ at time step t in this local-relation attention mechanism are represented as $\boldsymbol{V}^r_{att}$, the calculation formula is as follows:

$$\boldsymbol{V}^r_{att} = \sum_{i=1}^{N} \gamma_i^t (\boldsymbol{F}_i^l + \boldsymbol{F}_i^r) \tag{5}$$

γ_i^t is the attention weight of region R$_i$ at time step t, $\sum_i^N \gamma_i^t = 1$, which represents the focusing degree of each RoI of the image with its closely related RoIs. It is determined by the connection with the LSTM hidden layer information h_{t-1} at the previous time. The calculation method is as follows:

$$\gamma_i^t = softmax(\boldsymbol{W}_q^T tanh(\boldsymbol{W}_h h^{t-1} + \boldsymbol{W}_f(\boldsymbol{F}_i^l + \boldsymbol{F}_i^r) + \boldsymbol{b}_l)) \tag{6}$$

$\boldsymbol{W}_q$, $\boldsymbol{W}_h$, $\boldsymbol{W}_f$ and b_l are the parameters to be learned by training, which are shared by all functions in all time steps. The decoding process is as follow:

$$\begin{aligned} h_t^g &= LSTM_g([x_t; \boldsymbol{V}^g_{att}], h^g_{t-1}) \\ h_t^r &= LSTM_r([x_t; \boldsymbol{V}^r_{att}], h^r_{t-1}) \\ h_t^{out} &= Concat(h_t^g, h_t^r) \end{aligned} \tag{7}$$

As shown in Fig. 2, we concatenating the output hidden layer state h_t^g and h_t^r into h_t^{out} at timestep t, the probability vector $\boldsymbol{p}(y_t)$ of the next word is then calculated following

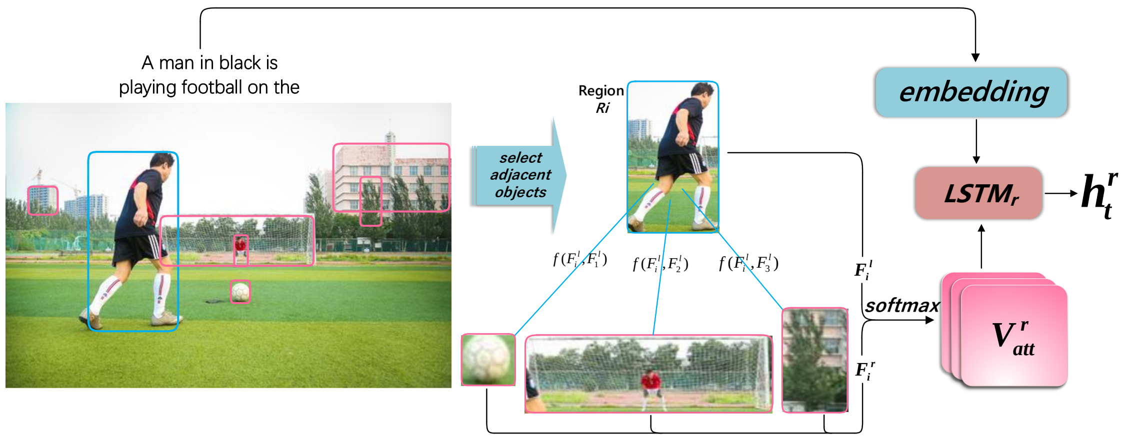

Figure 3: The illustration of calculating the relation feature of the object R_i, the weight $f(R_i^l, R_j^l)$ of every other K objects is obtained by dot-product and softmax operation. The local feature F_i^l and relation feature F_i^r of region R_i are packed together for attention weights distrubution.

the traditional LSTM operation in (8). So far, the output of the image caption generator is completed. We denote all parameters of GLRA as θ. In traditional MLE training, parameters θ are learned by minimizing the cross-entropy loss(XE) in (9). While in our model, the parameters θ are learned by self-critical adversarial training in SC-GAN and the MLE method is used to pre-train our generator.

$$p_\theta(y_t|I, y_{1:t-1}) = softmax(\boldsymbol{W}_p h_t^{out}) \tag{8}$$

$$L(\theta) = -\sum_{t=1}^{T} log(p_\theta(y_t|y_{1:t-1})) \tag{9}$$

3.2. Self-critical Generative Adversarial Network(SC-GAN)

Whether it is traditional cross-entropy training method or our self-critical adversarial training method, the goal is train a better θ-parameterized generative model G_θ. When meet reinforcement learning, the problem can be translated. In timestep t, the sequence $(y_1,...,y_{t-1})$ is denoted as state s, action a is the next selected word y_t, reward is the output of the discriminator, and the policy p_θ is decided by the generator $G_\theta(y_t|y_{1:t-1}, I)$. I is the input image features. After the next action is chosen, the state transition is determined. The φ-parameterized discriminative model D_φ is trained to provide a guidance for improving generator G_θ. $D_\varphi(Y_{1:T})$ is the probability represent how likely a sequence is ground truth or not. As shown in the right part of Fig. 1, D_φ is trained over the ground truth data and the generated data from G_θ. At the same time, the generative model G_θ is updated by using a policy gradient and Monte Carlo search on the basis of the expected end reward received from the D_φ. First of all, G_θ should be pre-trained on the sequence dataset s by

MLE method. Secondly, the same amount of generated samples and ground truth samples are transferred to D_φ for pre-training, then G_θ and D_φ will be trained alternately.

3.2.1. SC-GAN with Policy Gradient

We combine the GAN with th policy gradient algorithm. Because there is no intermediate reward, the goal of the generator(policy) G_θ is to generate a sequence from the initial state s_0 to maximize its expected end reward:

$$J(\theta) = \mathbb{E}[R_T|s_0,\theta] = \sum_{y_1\in\upsilon} G_\theta(y_1|s_0) * Q_{D_\varphi}^{G_\theta}(s_0,y_1) \tag{10}$$

where R_T is the reward for a complete sentence given by the discriminator D_φ. υ is the word dictionary. $G_\theta(y_1|s_0)$ is the probability of choosing y_1 as the next action. $Q_{D_\varphi}^{G_\theta}(s_0,y_1)$ is the action-value function. This formula means that at the first timestep with state s_0, the expected reward can obtained by adding the product of the probability of each action and their corresponding reward value. The objective of the generator is to optimize the θ to maximize the expected reward.

Since the problem is how to estimate the action-value function. We follow the traditional operation. Given a generated sequence, the model considers the estimated probability of being real given by $D_\varphi(Y_{1:T})$ as the reward. As shown in the following formula:

$$Q_{D_\varphi}^{G_\theta}(a=y_T, s=Y_{1:T-1}) = D_\varphi(Y_{1:T}) \tag{11}$$

Because the discriminator can only judge the complete sentence, as shown in right part of Fig.1, we adopt Monte Carlo search with a roll-out policy G_β to sample the future last T-t tokens. We represent an N-time search for each next word to evaluate the discriminator reward with this word. The N-time Monte Carlo search are represented as:

$$\{Y_{1:T}^1, ..., Y_{1:T}^N\} = \mathbf{MC}^{G_\beta}(Y_{1:t};N) \tag{12}$$

The $Y_{1:t}^n$ In this paper, the roll-out policy G_β is set the same as the generator G_θ. The $Q_{D_\varphi}^{G_\theta}(s=Y_{1:t-1}, a=y_t)$ is formulated as:

$$Q_{D_\varphi}^{G_\theta}(s=Y_{1:t-1}, a=y_t) = \begin{cases} \frac{1}{N}\sum_{n=1}^N D_\varphi(Y_{1:T}^n),\ Y_{1:T}^n \in MC^{G_\beta}(Y_{1:t};N) & \textit{for } t<T \\ D_\varphi(Y_{1:t}) & \textit{for } t=T \end{cases} \tag{13}$$

Now we have the the representation of each part of (10), the generator based on the policy G_θ intend to update the parameter θ to maxmize the long-term reward. Accordding to pg, the gradient of the objective function *J(θ)* in (10) to θ can be derived as:

$$\nabla_\theta J(\theta) = \sum_{t=1}^{T} \mathbb{E}_{Y_{1:t-1}\sim G_\theta}[\sum_{y_t\in\upsilon} \nabla_\theta G_\theta(y_t|Y_{1:t-1}) * Q_{D_\varphi}^{G_\theta}(Y_{1:t-1},y_t)] \tag{14}$$

The SC-GAN updates the generator G_θ by policy gradient algorithm too, but there are some differences. As we know the reward $Q_{D_\varphi}^{G_\theta}$ given by D_φ is a non-negative probability value.

Even if a worse result is generated, the discriminator will not punish the bad result, which will only reduce the probability of samples with less reward. However, due to uncontrollable factors such as sampling the padded token, the unclear reward and punishment system may make the training of the generator unfair. Therefore, the traditional greedy decoding algorithm is introduced to provide the baseline discriminator reward. As illustrated in the right part of Fig. 1. The greedy decoding method select the word with the highest probability at each timestep accordding to the $\boldsymbol{p(y_t)}$ generated by G_θ. We apply the G_θ trained in the last step to generate $\boldsymbol{p(y_t)}$. After greedy decoding finished, the D_φ will output this auxiliary sentence probability of being ground truth $D_\varphi(w^g_{1:T})$ and present it as the baseline reward. Thus our model start training in the form of self-critical adversarial. The greedy decoding process is as follows:

$$\begin{aligned} w^g_t &= \arg\max p(w_t|h^{out}_t) \\ r_{baseline} &= D_\varphi(w^g_{1:T}) \end{aligned} \tag{15}$$

The $Q^{G_\theta}_{D_\varphi}$ in (13) is supposed to be updated: each discriminator score of sentence sampled by N-time Monte Carlo search $D_\varphi(Y^n_{1:T})$ should subtract $D_\varphi(w^g_{1:T})$.

$$\hat{Q}^{G_\theta}_{D_\varphi}(s_{t-1}, y_t) = \begin{cases} \frac{1}{N}\sum_{n=1}^{N}(D_\varphi(Y^n_{1:T}) - D_\varphi(w^g_{1:T})),\ Y^n_{1:T} \in MC^{G_\beta}(Y_{1:t};N) & \mathit{for}\ t{<}T \\ \\ D_\varphi(Y_{1:t}) - D_\varphi(w^g_{1:T}) & \mathit{for}\ t{=}T \end{cases} \tag{16}$$

Since the expection can be estimated by sampling, the generator's parameters θ can be derived based on new action-value function in (16) as the following formula, referring to likelihood ratios, we further build an unbiased estimation on one episode:

$$\begin{aligned} \nabla_\theta J(\theta) &\simeq \sum_{t=1}^{T}\sum_{y_t\in\upsilon} \nabla_\theta G_\theta(y_t|Y_{1:t-1}) * \hat{Q}^{G_\theta}_{D_\varphi}(s_{t-1}, y_t) \\ &= \sum_{t=1}^{T}\sum_{y_t\in\upsilon} G_\theta(y_t|Y_{1:t-1}) * \nabla_\theta log\, G_\theta(y_t|Y_{1:t-1}) * \hat{Q}^{G_\theta}_{D_\varphi}(s_{t-1}, y_t) \\ &= \sum_{t=1}^{T} \mathbb{E}_{y_t\sim G_\theta(y_t|Y_{1:t-1})}[\nabla_\theta log\, G_\theta(y_t|Y_{1:t-1}) * \hat{Q}^{G_\theta}_{D_\varphi}(s_{t-1}, y_t)] \end{aligned} \tag{17}$$

As the expectation E can be approximated by sampling, we can update the generator's parameters as:

$$\theta \leftarrow \theta + a_h \nabla_\theta J(\theta) \tag{18}$$

Here a_h denotes the corresponding learning rate at step h. Once G_θ generates a more realistic sample, the model will retrain the discriminator D_φ according to the following formula:

$$\min_\varphi -\mathbb{E}_{Y\sim p(data)}[log\, D_\varphi(Y)] - \mathbb{E}_{Y\sim G_\theta}[log\,(1 - D_\varphi(Y))] \tag{19}$$

D_φ and G_θ are trained alternatively after pre-train stage. When G_θ has been trained for g-steps, the D_φ needs to be re-trained for d-steps to keep in good pace with G_θ, at each step in d, G_θ should provide different negative samples. The number of the positive samples from dataset S is set to the same as the negative samples from generator, with each pair of fused samples, we train D_φ for m epochs at each d step. The overall training process is shown in Algorithm.1.

Algorithm 1 Image Captioning Based on Self-critical Adversarial Training.

Input: generator policy G_θ; roll-out policy G_β; discriminator D_φ; a sequence dataset S.
Output: θ, φ
Initialize the G_θ and D_φ with random weights θ, φ
Pre-train G_θ on S on MLE
while *SC-GAN not converges* **do**
    for *g-steps* **do**
        Generate a sequence $Y_{1:T}$
        Compute baseline Discriminator score $D_\varphi(w^g_{1:T})$ based on G_θ at the last step
        for *t in 1:T* **do**
            Compute $Q(a = y_t, s = Y_{1:t-1})$ by (16)
        end
        Update generator parameters by (18)
    end
    for *d-steps* **do**
        Use current G_θ to generate negative samples and combine with ground truth one
        Train Discriminator D_φ for m epochs by (19)
    end
    $\beta \leftarrow \theta$
end

3.3. The Discriminative Model

The purpose of the discriminator is to classify the sequence correctly. The popular discriminators are deep neural network(DNN), convolutional neural network(CNN), and recurrent convolutional neural network(RCNN). In the SC-GAN, we choose the CNN as our discriminator. We focus on the situation where the discriminator predicts the probability that a finished sequence is real. Firstly we represent an input sequence $x{-}1$,...,x_T as:

$$\xi_{1:T} = \boldsymbol{x}_1 \oplus \boldsymbol{x}_2 \oplus ... \oplus \boldsymbol{x}_T \tag{20}$$

Where $\boldsymbol{x}_T \in R^k$ is the word embedding and $\oplus$ is the concatenation operation. Afterward a $\boldsymbol{W}_d \in \mathrm{R}^{T\times K}$ apply a convolutional operation to a window size of l words to produce a new feature map:

$$c_i = \rho(\boldsymbol{W}_d \otimes \xi_{i:i+l-1} + b) \tag{21}$$

$\otimes$ denotes the operation of elementwise production, ρ is the non-linear function. Then we select the max one of all c_i, $\tilde{c_i}$=max$\{c_1$,...,$c_{T-l+1}\}$.

Finallya fully connected layer with sigmoid activation is used to output the probability that the input sequence is real. We update the parameter φ by minimize the cross entropy between the ground truth token and the predicted probability as formula in (19). The G_θ and D_φ are trained altenately after the pre-train phase.

4. Experimental Results and Analysis

4.1. Implementation Details

We use the popular MSCOCO dataset to validate the performance of the proposed method. In the phase of extracting global features, we adapt ResNet-101 without the last two layers, and fine-tune their parameters on the MSCOCO. The extracted image feature $\boldsymbol{I}$ has a fixed size of 2048*14*14, so the parameter n in GLRA is 14 C is 2048 and L is 256. The C' in $\boldsymbol{W}_q$ and $\boldsymbol{W}_k$ is 64 and C" in $\boldsymbol{W}_v$ is 512. In more details, the number of neurons in LSTM_g and LSTM_r sets to 512. The attention weights α has the same space size with $\boldsymbol{V}$, which is 14*14. We also retrieve local object features using a Faster R-CNN pre-trained on the MSCOCO dataset. The parameter N in the local-realtion attention is 30 so the top-30 detected object features are selected to calculate the relation features. To determine the best number of adjacent object feature for each R_i, we conduct an ablation study with different choice of K. The result is shown in TABLE. 1. With the increase of number K from 5 to 15, the performance becomes better. Finally, the number of K sets to 15 to explore the relationship as sufficient as possible.

Following the optimal parameters setting in SeqGAN, the g,d and m in SC-GAN are set as 1, 5 and 3 separately and the maximum length of input sentence is set to 20. We firstly pre-train the G_θ for 100 epochs by MLE and subsequently pre-train the D_φ until it converges. Then the G_θ and D_φ can follow the adversarial training scheme. The batch size is set to 32 and learning rating is 0.001. All experiments are conducted on a server embedded with NVIDIA RTX2080Ti GPU and Ubuntu16.04 system.

Table 1: Performance comparison with different K-number adjacent objects in the Local-Relation Attention. All experiments are ensembled with SC-GAN.

Methods	**B@1**	**B@4**	**M**	**R**	**C**	**S**
GLRA with 5 Objects for Local-Relaiton Attention	79.6	39.4	25.9	56.5	126.3	21.7
GLRA with 10 Objects for Local-Relaiton Attention	81.9	41.2	28.8	58.6	128.6	23.1
GLRA with 15 Objects for Local-Relaiton Attention	**82.5**	**41.7**	**29.6**	**60.1**	**131.6**	**23.9**
GLRA with 20 Objects for Local-Relaiton Attention	82.1	41.3	29.3	59.1	130.1	23.3

4.2. Result and Analysis

4.2.1. Ablation Experiments

In order to independently verify the effectiveness of GLRA, we first integrate the traditional MLE training method to conduct experiments. Compared with other advanced models that also used the cross-entropy method for training, the experimental results are shown in TABLE 2. What stands out in the table is that the GLRA with the cross-entropy loss training method brings improvement in the major metric, which proves that it can make more reasonable use of the image feature information and excavate the potential internal relationship of the image regions.

To verify the effectiveness of SC-GAN, we further combine it with the GLRA and compare the performance with the model that only contains GLRA. The improvement is evident by comparing the results of TABLE 2 and TABLE 3. In addition, we also conduct comparative experiments with some advanced RL-based methods to verify the capacity of the whole model. As can be seen from the TABLE 3, when the SC-GAN is combined with GLRA, there is a more significant increment in most metrics, our model can also bring comparable even better results compared with start-of-the-art methods in recent years, including several prevailing transformer-based models.

Table 2: Performance of our model and other advanced models based on cross-entropy, where B@N, M, R, C and S are short for BLEU@N, METEOR, ROUGE-L, CIDEr-D and SPICE scores.

Methods	**B@1**	**B@4**	**M**	**R**	**C**	**S**
SCSTRennie et al. (2017)	-	30.0	25.9	53.4	99.4	-
HANWang et al. (2019)	77.2	36.2	27.5	56.6	114.8	20.6
DAIC Wei et al. (2020)	73.7	34.2	26.4	54.8	106.2	-
Up-Down Anderson et al. (2018)	77.2	36.2	27.0	56.4	113.5	20.3
RFNet Jiang et al. (2018)	76.4	35.8	27.4	56.8	112.5	20.5
GCN-LSTM Yao et al. (2018)	77.3	36.8	27.9	57.0	116.3	20.9
AoANet Huang et al. (2019)	77.4	37.2	28.4	57.5	119.8	21.3
ARL Wang et al. (2020a)	75.9	35.8	27.8	56.4	111.3	-
CL-topdown Wang et al. (2020c)	-	37.1	27.9	57.2	117.1	-
X-Linear Pan et al. (2020)	77.3	37.0	28.7	57.5	**120.0**	21.8
Ours	**78.3**	**37.9**	**28.9**	**58.5**	119.6	**21.9**

4.2.2. Qualitative Analysis

In order to show our model's effect more intuitively, we visualize the attention weights in Fig. 4 to demonstrate that our model can accurately simulate human perception. We first expand our attention weight 24 times and adjust it to the same size as the input image by the Gaussian filter. Closer inspection of Fig. 4 shows that the model can not only focus on the corresponding target image area when generating the main object, but also grasp the key areas in the graph when generating the words describing the relationship between different objects. For example, in Fig. 4 (a), when generating the word "riding", the model obviously focuses on the image part connected to the person and the motorcycle. In Fig. 4 (c), when generating the word "baseball", the image not only pays attention to the word "baseball" itself, but also pays adequate attention to the baseball cap on the head. These demonstrates the model can utilize the semantic information effectively.

The effect of our model at the sentence level is presented in Fig. 5, we compare the ground-truth sentences, descriptions generated by the HAN Wang et al. (2019) model with reinforcement learning, since it also organize a hierarchical attention mechanism, and the

Table 3: Performance comparison with other advanced models based on Reinforcement Learning. † means the original model is ensembled with self-critical training.

Methods	B@1	B@4	M	R	C	S
SCST:Att2all Rennie et al. (2017)	-	34.2	26.7	55.7	114.0	-
DAIC† Wei et al. (2020)	77.6	35.4	26.7	56.5	116.8	-
HAN† Wang et al. (2019)	80.9	37.6	27.8	58.1	121.7	21.5
UP-DOWN† Anderson et al. (2018)	79.8	36.3	27.7	56.9	120.1	21.4
GCN-LSTM† Yao et al. (2018)	80.5	38.2	28.5	58.3	127.6	22.0
IIEK† Huang et al. (2020)	79.3	37.3	27.7	56.9	120.4	-
IDGAN Liu et al. (2020)	81.3	38.5	28.5	58.8	123.5	-
AoANet†Huang et al. (2019)	81.6	40.2	29.3	59.4	132.0	22.8
SLL-SLEChen and Jin (2020)	-	-	27.0	-	119.6	19.9
POS-SCAN Zhou et al. (2020)	80.2	38.0	28.5	-	126.1	22.2
X-Linear† Pan et al. (2020)	80.9	39.7	29.5	59.1	**132.8**	23.4
M2 Transformer Cornia et al. (2020)	80.8	39.1	29.2	58.6	131.2	22.6
Ours	**82.5**	**41.7**	**29.6**	**60.1**	131.6	**23.9**

sentences by our model. The red texts are the sentences generated by the proposed model, which are more accurate and natural than the HAN model, which are shown in blue. Significantly, the proposed model shows superior performance in detecting the fine-grained properties of the image. For example, in Fig. 5 (c), we successfully detect the "barrel", and in (d) the keyword "ball" is obtained. What's more, we successfully excavate the critical relationship between image areas. In Fig. 5 (a), the successful detection of the verb "riding" shows the importance of relation features. Besides, it is believed that the word "barrel" plays a crucial role in generating the word "wine", which indicated our local-relation attention mechanism could effectively take advantage of the potential information between regions again. Our model has impressive performance in generating the words that describe the regional relationship to obtain a more vivid and appropriate image caption.

5. Conclusion

In this paper, we propose a new fused attention mechanism, integrating global attention achieved by self-attention and local-relation attention. For each region, the relation features are assigned with attention weights together with the local features to better excavate the potentially important information of the image. Besides, we also improve the traditional GAN with a self-critical training method. In this way, the reward and punishment system becomes more explicit. The model training process can be more stable and effective. Experiments on the MSCOCO dataset demonstrate both of the two innovations can boost the quality of the generated sentences.

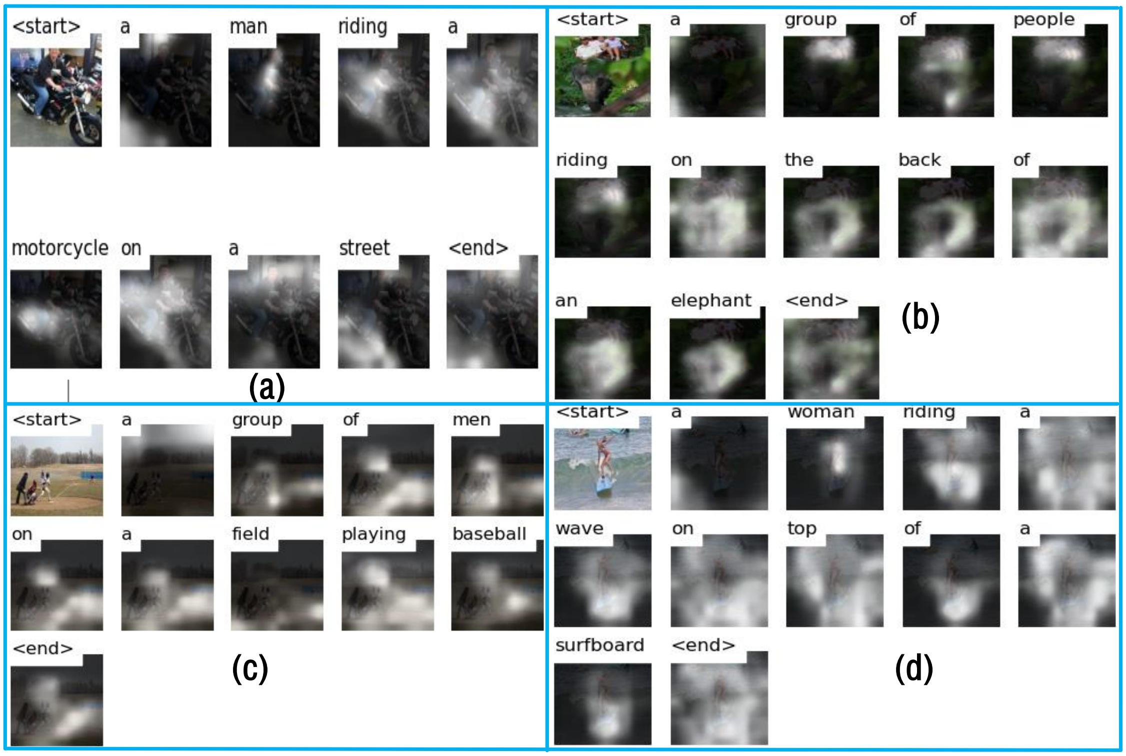

Figure 4: Examples illustrate word prediction when attending on different image regions.

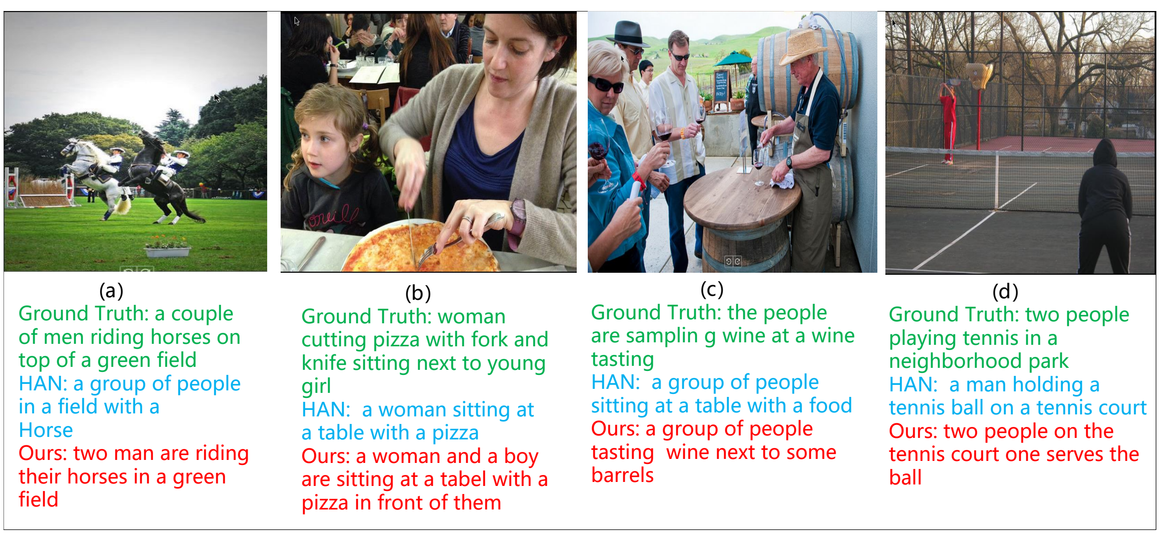

Figure 5: Visualization of the generated descriptions. All samples are randomly selected.

Acknowledgments

This work is supported by National Natural Science Foundation of China(Nos. 61966004, 61866004, 61762078), Guangxi Natural Science Foundation(Nos. 2019GXNSFDA245018, 2018GXNSFDA281009), Innovation Project of Guangxi Graduate Education (No.XYCBZ20 -21002), Guangxi "Bagui Scholar" Teams for Innovation and Research Project, Guangxi Talent Highland Project of Big Data Intelligence and Application, and Guangxi Collaborative Innovation Center of Multi-source Information Integration and Intelligent Processing.

References

Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 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*, pages 6077–6086, 2018.

Jia Chen and Qin Jin. Better captioning with sequence-level exploration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10890–10899, 2020.

Marcella Cornia, Matteo Stefanini, Lorenzo Baraldi, and Rita Cucchiara. Meshed-memory transformer for image captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10578–10587, 2020.

Antonia Creswell, Tom White, Vincent Dumoulin, Kai Arulkumaran, Biswa Sengupta, and Anil A Bharath. Generative adversarial networks: An overview. *IEEE Signal Processing Magazine*, 35(1):53–65, 2018.

Feicheng Huang, Zhixin Li, Haiyang Wei, Canlong Zhang, and Huifang Ma. Boost image captioning with knowledge reasoning. *Machine Learning*, 109(12):2313–2332, 2020.

Lun Huang, Wenmin Wang, Jie Chen, and Xiao-Yong Wei. Attention on attention for image captioning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4634–4643, 2019.

Wenhao Jiang, Lin Ma, Yu-Gang Jiang, Wei Liu, and Tong Zhang. Recurrent fusion network for image captioning. In *Proceedings of the European Conference on Computer Vision*, pages 499–515, 2018.

Junhao Liu, Kai Wang, Chunpu Xu, Zhou Zhao, Ruifeng Xu, Ying Shen, and Min Yang. Interactive dual generative adversarial networks for image captioning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 11588–11595, 2020.

Yingwei Pan, Ting Yao, Yehao Li, and Tao Mei. X-linear attention networks for image captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10971–10980, 2020.

Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. Sequence level training with recurrent neural networks. *arXiv preprint arXiv:1511.06732*, 2015.

Steven J Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. Self-critical sequence training for image captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7008–7024, 2017.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008, 2017.

Junbo Wang, Wei Wang, Liang Wang, Zhiyong Wang, David Dagan Feng, and Tieniu Tan. Learning visual relationship and context-aware attention for image captioning. *Pattern Recognition*, 98:107075, 2020a.

Li Wang, Zechen Bai, Yonghua Zhang, and Hongtao Lu. Show, recall, and tell: Image captioning with recall mechanism. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 12176–12183, 2020b.

Weixuan Wang, Zhihong Chen, and Haifeng Hu. Hierarchical attention network for image captioning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8957–8964, 2019.

Ziwei Wang, Zi Huang, and Yadan Luo. Human consensus-oriented image captioning. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, pages 659–665, 2020c.

Haiyang Wei, Zhixin Li, Canlong Zhang, and Huifang Ma. The synergy of double attention: Combine sentence-level and word-level attention for image captioning. *Computer Vision and Image Understanding*, 201:103068, 2020.

Ting Yao, Yingwei Pan, Yehao Li, and Tao Mei. Exploring visual relationship for image captioning. In *Proceedings of the European Conference on Computer Vision*, pages 684–699, 2018.

Quanzeng You, Hailin Jin, Zhaowen Wang, Chen Fang, and Jiebo Luo. Image captioning with semantic attention. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4651–4659, 2016.

Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. Seqgan: Sequence generative adversarial nets with policy gradient. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

Yuanen Zhou, Meng Wang, Daqing Liu, Zhenzhen Hu, and Hanwang Zhang. More grounded image captioning by distilling image-text matching model. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4777–4786, 2020.