FFGS: Feature Fusion with Gating Structure for Image Caption Generation

Aihong Yuan[1,2], Xuelong Li[1], and Xiaoqiang Lu[1(✉)]

[1] Center for OPTical IMagery Analysis and Learning (OPTIMAL), Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an 710119, Shaanxi, People's Republic of China
{ahyuan,xuelong_li}@opt.ac.cn, luxq666666@gmail.com
[2] University of Chinese Academy of Sciences, 19A Yuquanlu, Beijing 100049, People's Republic of China

Abstract. Automatically generating a natural language to describe the content of the given image is a challenging task in the interdisciplinary between computer vision and natural language processing. The task is challenging because computers not only need to recognize objects, their attributions and relationships between them in an image, but also these elements should be represented into a natural language sentence. This paper proposed a feature fusion with gating structure for image caption generation. First, the pre-trained VGG-19 is used as the image feature extractor. We use the FC-7 and CONV5-4 layer's outputs as the global and local image feature, respectively. Second, the image features and the corresponding sentence are imported into LSTM to learn their relationship. The global image feature is gated at each time-step before imported into LSTM while the local image feature used the attention model. Experimental results show our method outperform the state-of-the-art methods.

Keywords: Image caption generation · Recurrent neural network
Convolutional neural network · Multi-modal embedding · Feature fusion

1 Introduction

Image caption, which automatically generates a natural language sentence to describe the content of the given image, has recently become a challenging but fundamental task of *computer vision* (CV) and *natural language processing* (NLP) [3,7,13,19]. The task is challenging because the caption generation models not only should solve the computer vision challenges of determining what objects are in an image [11,22], but also be powerful enough to describe their relationships with natural language. However, challenges and opportunities coexist. Image caption links image and natural language together, which makes it has a great potential for application in the near future. Therefore, many researchers have paid great attention to this task.

J. Yang et al. (Eds.): CCCV 2017, Part I, CCIS 771, pp. 638–649, 2017.
https://doi.org/10.1007/978-981-10-7299-4_53

Approaches for image caption generation task have two main categories: (1) retrieval-based methods and (2) *multi-modal neural networks-based* (MMNN-based) methods. Before the advent *deep learning* (DL), retrieval-based methods are the most popular methods for image caption generation. These methods retrieval similar objects and then retrieval a similar sentence from the training dataset [6,16]. After that, the words are connected together according to certain grammar rules.

Although the retrieval-based methods have gained many encouraged results, many problems have not been solved. These methods need fixed visual concepts and hard-coded sentence template, which makes the sentences generated by these models are less variety.

Recently, deep learning has achieved many breakthroughs in *natural language processing* (NLP) and *computer vision* (CV), such as *machine translation* (MT) [2], image classification, object detection [15,17], *etc.* The main success of deep learning is that the ability of representation is powerful. Some image caption generating methods using deep neural networks have been proposed recently. For example, *multi-modal Recurrent networks* (m-RNN) is proposed by Mao *et al.* [12], which uses the *convolutional neural networks* (CNNs) as feature extractor and the traditional *recurrent neural networks* (RNNs) as sentence generator. From then on, the "CNN + RNN" mode becomes the most popular scheme for image caption generation.

Compared to the retrieval-based methods, the *multi-modal neural network-based* (MMNN-based) methods have shown a greater improvement on the performance of image caption generating task. Sentence generated by the MMNN-based methods is more reasonable and more changeable. However, they also have some shortages. For example, m-RNN [12], Google-NIC [18], and LRCN [4] only used the global image feature vector from the fully connected layer of the CNN, which can not dig up the subtle relationship between the image and the natural statement. On the contrary, NIC-VA [20] only uses the local image feature, which may lead to loss the global information of the image.

To overcome these shortages, we propose a *feature fusion method with gating structure* (**FFGS**). Global image features are added on the basis of the NIC-VA. The global image features are imported into the sentence generator at each time-step. Unlike the m-RNN, Google-NIC, and LRCN, which import the global image feature into RNN at the first time-step or at each time-step without any processing, we use gating mechanism for the global image features. In other words, the global image features are gated at each time-step before imported into the RNN unit.

The main contributions of the proposed algorithm are as follows:

- Feature fusion strategy is used in this work. The proposed algorithm uses the global and local image features to guarantee the information of images be more comprehensively and meticulously used.
- Gating mechanism is used for the global image feature. We use gate to control the global image feature and this mechanism can solve the argument whether should import the global image feature into the sentence generator at the

first time-step or at each time-step. Furthermore, it robustly solve how much should be imported at each time-step.
– The prosed algorithm is tested on three benchmark datasets. The experimental results show that method proposed in this paper is better than the state-of-art methods.

The rest of this paper is organized as follows. In Sect. 2, some previous works are briefly introduced. Section 3 presents our model for image caption generation. To validate the proposed method, the experimental results are shown in Sect. 4. At last, Sect. 5 makes a brief conclusion for this paper.

2 Related Work

2.1 Deep Neural Networks for MT

Recently, many works showed that deep neural network can be successfully used to solve lots of problems in NLP, such as *machine translation* (MT). In the conventional MT system, the neural network refers to as an RNN Encoder-Decoder which consists of two RNN [2]. One acts as an encoder which maps a variable-length source sentence to a fixed-length vector. And the other acts as a decoder which decodes the vector produced by the encoder into a variable-length target sentence. When RNN trained with *Backpropagation Through Time* (BPTT) [5], there exists some difficulties in learning long-term dependency due to the so-called vanishing and exploding gradient problems. To overcome these difficulties, some gated RNNs (*e.g.* LSTM and GRU) have been proposed. Some researchers treat image caption generation as a machine translation problem. However, the input is image which is not a sequence signal, so they use pre-trained CNN as encoder for image, instead encoding RNN in MT. The proposed method in this paper also follows this idea.

2.2 Generating Sentence Descriptions for Images

There are mainly two categories of methods for image caption generation task. The first category is retrieval-based methods which retrieve similar captioned images and generate new descriptions by retrieving a similar sentence from a image-description dataset. These methods project image and sentence representations into a common semantic space, which is used for ranking image captions or for image search. They retrieve similar objects and the corresponding descriptions from the training data, and then stitch these descriptions into sentences. A typical work is called BabyTalk system [9] which consists of two important steps. The first step is content planing, which is detecting the defined objects and its corresponding content words. At the second step, put these words obtained in the first step into a sentence in accordance with certain rules. Socher *et. al* propose a method named DT-RNN [16] to generate description for a given image. Dependency trees are used to embed sentences into a vector space aims to retrieve corresponding images. Another typical category is multi-modal neural network

based methods. These methods based on multi-modal embedding models, generated sentence in a word-by-word manner and conditioned on image representation which is the output from a deep convolutional network. Our method falls into this category and our multi-modal embedding model is a recurrent network. Our model is trained to maximize the likelihood of the target sentence conditioned on the given training image. Some previous works seem closely related to our method such as m-RNN [12], Google-NIC [18], LRCN [4] and NIC-VA [20]. However, the proposed model in this paper uses the global and local feature fusion strategy, which is different from the aforementioned models.

3 Proposed Method

In this section, we introduce our whole model named **FFGS** (*i.e.* Feature Fusion with Gating Structure). Figure 1 shows our full model diagram and it main contains two modules: (1) image representation and (2) multi-modal embedding. VGG-19 is used for image representation. LSTM is used for multi-modal embedding. In other words, image information and the corresponding sentence are embedding in LSTM. Now, we introduce our FFGS in detail.

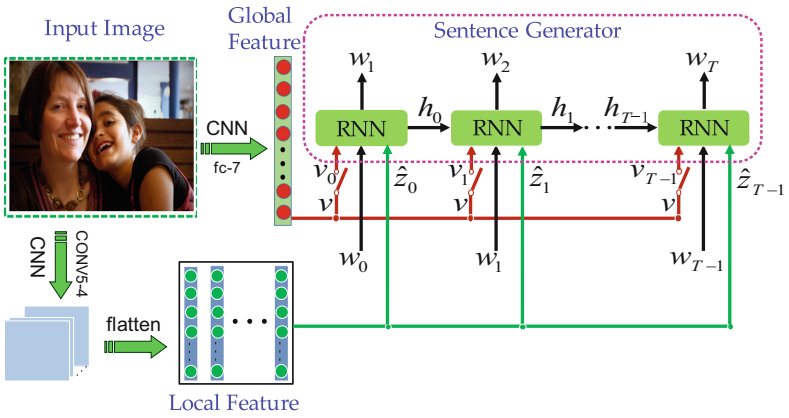

Fig. 1. Overview of our method for image caption generation. A deep CNN for image features extraction: the FC7 layer is used to extract global features and the CONV5-4 layer is used to extract local features of the given images. RNN model acts as a decoder which decodes image features into sentences. The gate for controlling the global image feature is computed with the pre-step hidden state of RNN. The local image feature are selected by the pre-step hidden state and the current word vector.

3.1 Image Feature Representation

As Fig. 1 is shown, our image feature representation concludes two parts: global feature and local feature. We use the feature vector output from FC-7 and CONV5-4 layers of VGG-19 as the global and local image feature, respectively.

Image Global Representation: The VGG-19 is pre-trained on ImageNet and used as the image encoder in our model. The global representation of image I is as follows:

$$\mathbf{v} = \mathbf{W}_I \cdot [Fc(I)] + \mathbf{b}_I, \tag{1}$$

where I donates the image I, $Fc(I) \in \mathbb{R}^{4096}$ is the output of the FC-7 layer. The matrix $\mathbf{W}_I \in \mathbb{R}^{h \times 4096}$ is a embedding matrix which projects 4096-dimension image feature vectors into the embedding space with h-dimension and $\mathbf{b}_I \in \mathbb{R}^h$ donates the bias. $\mathbf{v} \in \mathbb{R}^h$ is so-called image global feature representation because it is computed with the entire image I.

Image Local Feature Representation: The VGG-19 also be used as image local feature extractor in our model. When a raw image $\mathrm{I} \in \mathbb{R}^{W \times H \times 3}$ is input to VGG-19, the CONV5-4 layer outputs feature map $\mathbf{v}_c \in \mathbb{R}^{W' \times H' \times D}$. Then, we flatten this feature map into $\mathbf{v}_l \in \mathbb{R}^{D \times C}$, where $C = W' \times H'$. This processing program can be written as follows:

$$\mathbf{v}_l = \{\mathbf{v}_{l1}, \mathbf{v}_{l2}, \cdots, \mathbf{v}_{lC}\} = flatten\left(Conv(I)\right), \tag{2}$$

where $\mathbf{v}_{li} \in \mathbb{R}^D,\ i \in \{0, 1, \cdots, C\}$ donates the feature of i-th location of image I. In other words, each image I is divided into C locations and every $\mathbf{v}_{li}$ represents one location. So, $\mathbf{v}_{li}$ is the location feature representation.

3.2 Sentence Representation

In our model, we encode words into one-hot vectors. For example, the benchmark dataset has N_0 different words, every word is encoded into N_0-dimension vector which only one value equals to 1 and others equal to 0. When a raw image import into our model, a corresponding sentence S is generated which is encoded as a sequence of one-hot vectors. We donate $S = (\mathbf{w}_1, \mathbf{w}_2, \cdots, \mathbf{w}_T)$, where $\mathbf{w}_i \in \mathbb{R}^{N_0}$ donates the i-th word in the sentence. We embed these words into embedding space. The concrete formula is as follows:

$$\mathbf{s}_t = \mathbf{W}_s \cdot \mathbf{w}_t,\ t \in \{1, 2, \cdots, T\}, \tag{3}$$

where $\mathbf{W}_s$ is the embedding matrix of sentences which projects the word vector into the embedding space. So the projection matrix $\mathbf{W_s}$ is a $h \times N_0$ matrix where N_0 is the size of the dictionary and h is the dimension of the embedding space.

3.3 LSTM for Sentence Generating

LSTM is used as sentence generator in our model. In other words, RNN model in Fig. 1 is LSTM. As illustrated in Fig. 2, every LSTM unit has four inputs: the local image feature $\mathbf{v_t}$, the global image feature $\hat{\mathbf{z}}_t$, the word $\mathbf{s_t}$ and the previous hidden state $\mathbf{h_{t-1}}$.

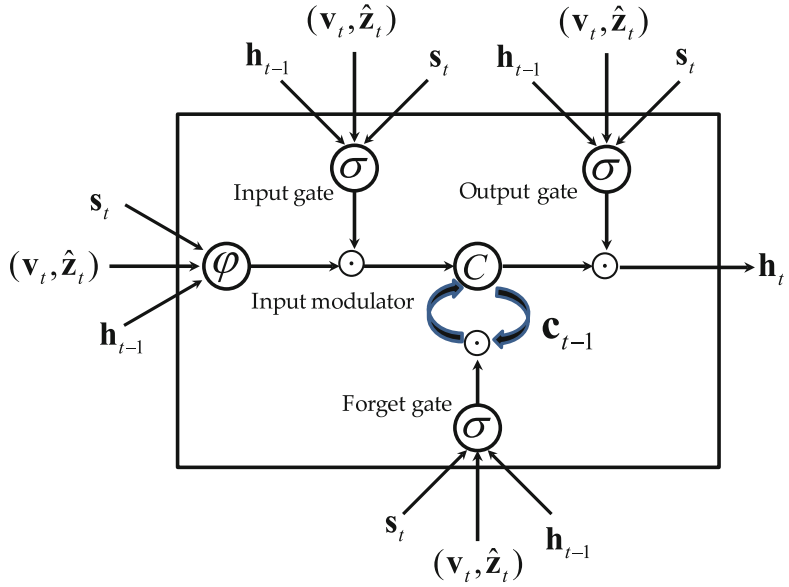

Fig. 2. The diagram of the LSTM unit used in our model. Each gate has 4 input vectors: the global feature at the time-step t $\mathbf{v}_t$, the local image feature at the time-step t $\hat{\mathbf{z}}_t$, word representation at the time-step t $\mathbf{s}_t$ and the previous hidden state $\mathbf{h}_{t-1}$.

LSTM Model: In this subsection, we introduce our formula in detail. First, three gates and updating memory content of LSTM are rewritten as follows:

$$\mathbf{i}_t = \sigma\left(W_i\mathbf{s}_t + U_i\mathbf{h}_{t-1} + V_i\mathbf{v}_t + Z_i\hat{\mathbf{z}}_t + \mathbf{b}_i\right), \tag{4}$$

$$\mathbf{f}_t = \sigma\left(W_f\mathbf{s}_t + U_f\mathbf{h}_{t-1} + V_f\mathbf{v}_t + Z_f\hat{\mathbf{z}}_t + \mathbf{b}_f\right), \tag{5}$$

$$\mathbf{o}_t = \sigma\left(W_o\mathbf{s}_t + U_o\mathbf{h}_{t-1} + V_o\mathbf{v}_t + Z_o\hat{\mathbf{z}}_t + \mathbf{b}_o\right), \tag{6}$$

$$\tilde{\mathbf{c}}_t = \tanh\left(W_c\mathbf{s}_t + U_c\mathbf{h}_{t-1} + V_c\mathbf{v}_t + Z_c\hat{\mathbf{z}}_t + \mathbf{b}_c\right), \tag{7}$$

where $W_* \in \mathbb{R}^{h\times h}$, $U_* \in \mathbb{R}^{h\times h}$, $V_* \in \mathbb{R}^{h\times h}$ and $Z_* \in \mathbb{R}^{h\times h}$ are donate weights matrixes and $\mathbf{b}_* \in \mathbb{R}^h$ donate biases. $\mathbf{v_t} \in \mathbb{R}^h$ and $\hat{\mathbf{z}}_t \in \mathbb{R}^h$ are the global and the local image features, respectively. Their calculating formulas are introduced in the next two subsections.

The current memory and hidden state are computed as follows:

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \tilde{\mathbf{c}}_t, \tag{8}$$

$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t), \tag{9}$$

The LSTM module outputs a probability at each time-step. We write it as following formula:

$$\mathbf{p}_{t+1} = \text{softmax}(\mathbf{y}_t) = \text{softmax}(W_y\mathbf{h}_t + \mathbf{b}_y), \tag{10}$$

where $W_y \in \mathbb{R}^{N_0 \times h}$ and $\mathbf{b}_y \in \mathbb{R}^{N_0}$ donate passing forward parameters. $\mathbf{y}_t \in \mathbb{R}^{N_0}$ is a output of LSTM at the t-th time-step. $\mathbf{p}_{t+1} \in \mathbb{R}^{N_0}$ is a probability vector whose each element donates the predicting probability of the corresponding word.

Having builded the LSTM model, initializing the system is another important thing to do. The memory and the hidden state is initialized by the following formulas:

$$\mathbf{c}_0 = \tanh\left(W_{c\_init}\left(\frac{1}{C}\sum_{i=1}^{C}\mathbf{v}_{li}\right) + \mathbf{b}_{c\_init}\right), \tag{11}$$

$$\mathbf{h}_0 = \tanh\left(W_{h\_init}\left(\frac{1}{C}\sum_{i=1}^{C}\mathbf{v}_{li}\right) + \mathbf{b}_{h\_init}\right), \tag{12}$$

where $W_{c\_init} \in \mathbb{R}^{h \times h}$ and $W_{h\_init} \in \mathbb{R}^{h \times h}$ are initial weights. $\mathbf{b}_{c\_init} \in \mathbb{R}^h$ and $\mathbf{b}_{h\_init} \in \mathbb{R}^h$ are initial biases.

Gate for Global Image Feature: $\mathbf{v}_t$ occurs several times in LSTM model. And it is a output of global image feature controlled by gate. In previous works, the vast majority of them import the global feature defined in Eq. (1) at the first time-step or at each time-step into the RNN decoder, but they find that global feature imported at the first time-step is better than at every time-step. They explain that global feature imported at each time-step may bring more noise to the system. However, this reason can not convince us. So we want to design a robust algorithm that able to autonomously decide how many global feature should be imported into the decoder. Inspired by the gate technology exploited in RNN, we design a gate before the global feature imported into the system. The gate is defined as follows:

$$g_t = \sigma(\mathbf{w}_g^T \mathbf{h}_{t-1} + b_g), \tag{13}$$

where $\mathbf{w}_g \in \mathbb{R}^h$ is weight vector, b_g is bias. So the t-th gate g_t is a scaler and its value correlates with the previous time-step hidden state $\mathbf{h}_{t-1}$.

After calculating the gate, the global image feature at time-step t is computed as follows:

$$\mathbf{v}_t = g_t \mathbf{v}. \tag{14}$$

Through Eq. (14), if we set $g_t = 1$ at $t = 0, 1, \cdots, T$, $\mathbf{v}$ is imported into the decoder at each time-step. If we set $g_t = 1$ at $t = 0$ and $g_t = 0$ at $t = 1, 2, \cdots, T$, $\mathbf{v}$ is only imported into the decoder at the first time-step. So introduced gate concept, intuitively feel our algorithm is robust for global image feature and methods in previous works are special cases of our approach.

Attention Mechanism for Local Image Feature: The local image feature $\hat{\mathbf{z}}_t$ donates the local information of image. Here we use attention mechanism as introduced in references [20] for local feature. At each time-step, the attention

mechanism uses the previous hidden state h_{t-1} to decide the local feature. The attention model is defined as follows:

$$\alpha_t = \text{softmax}\left(\tanh\left[\left(\mathbf{w}_a^T \mathbf{v}_l\right)^T + U_a \mathbf{h}_{t-1} + \mathbf{b}_a\right]\right) \triangleq \left[\alpha_{t1} \cdots \alpha_{tC}\right]^T, \tag{15}$$

where $\mathbf{w}_a \in \mathbb{R}^h$ and $U_a \in \mathbb{R}^{h \times h}$ are weights. $\mathbf{v}_l$ is defined in Eq. (2). $\mathbf{b}_a \in \mathbb{R}^h$ is bias. $\alpha_t \in \mathbb{R}^C$ is a probability vector whose each dimension value donates the probability of the corresponding local image feature. In our algorithm, we use the soft attention model. Therefore, $\hat{\mathbf{z}}_t$ is calculated as follows:

$$\hat{\mathbf{z}}_t = \mathbf{v}_l \alpha_t = \sum_{i=1}^{C} \alpha_{ti} \mathbf{v}_{li}. \tag{16}$$

Through Eq. (16) we know that α_t decides which locals should be used at the current time-step.

The loss function of our model can be written as the negative likelihood function, which formula is as follows:

$$L(\theta) = -\log P\left(S \,|I\right) + \lambda_\theta \|\theta\|^2 + \lambda_\alpha \left(1 - \sum_{i=1}^{C} \alpha_{ti}\right)^2, \tag{17}$$

where θ is all parameters set which concludes parameters of the LSTM, embedding matrixes in Sects. 3.1 and 3.2 and all gate models. $\lambda_\theta \cdot \|\theta\|_2^2$ is a regularization term. $\lambda_\alpha \left(1 - \sum_{i=1}^{C} \alpha_{ti}\right)^2$ is a probabilistic constraint.

The proposed model is trained with *back-propagation through time* (BPTT) algorithm to minimize the cost function $L(\theta)$.

4 Experimental Evaluation

In this section, we describe our experimental methodology and quantitative results which validate the effectiveness of our model for caption generation.

4.1 Datasets and Data Processing

Datasets. The Flickr8K [14], Flickr30K [21] and MS COCO [10] datasets are used in our experiments. Flickr8K focus on activities of people and animals (mainly dogs) and it contains almost 8,000 images and each image contain 5 corresponding descriptions. Flickr30K is a extension of Flickr8K within almost 30,000 images. Recently, MS COCO is the biggest and most challenging dataset for image caption generation. It contains 82,783 training images, 40,504 validation images and 40,775 testing images.

Data Processing. Flickr8K and Flickr30K do not have clearly training, validation and testing sets. So we choose 1,000 images for validation and 1,000

testing and the rest for training from all the datasets, which is same as reference [8]. Though MS COCO have clearly training, validation and testing sets, but the testing set does not have sentence descriptions for images, so we randomly extract both 5,000 images and their corresponding descriptions from the verification set as validation and testing data. Unlike Flickr8K and Flickr30K, some images in MS COCO having more than 5 describing sentences. To grantee each image has the same number caption, we discard data which caption in excess of 5. For all our experiments, we use a fixed vocabulary size of 10,000.

4.2 Evaluation Metrics

In order to evaluate the proposed method, three objective metrics are used in this paper. They are BLEU and METEOR. BLEU score represents the precision ratio of the generated sentence compared with the reference sentences. METEOR score reflects the precision and recall ratio of the generated sentence. It is based on the harmonic mean of uniform precision and recall.

4.3 Quantitative Evaluation and Analysis

Table 1 shows the generation results on the three standard datasets compared with the most typical and state-of-the-art models. The Results show that our model outperforms all the other models. Among them, m-RNN and LRCN use the output of the CNN FC layer as the image feature (*i.e.* image global feature in our model). The image feature is imported into the RNN unit at each time-step. The two models have a little difference which is the language model—m-RNN uses the "vanilla" RNN but LRCN uses LSTM use as the sentence generator. Different from the two aforementioned methods, DeVS [8] and Google-NIC import the whole image feature into RNN only at the first time-step. Through the results we can know that Google-NIC shows a better performance than DeVS. The main reason is that Google-NIC uses LSTM as language model which is much better than the "vanilla" RNN which DeVS used. As compared models, NIC-VA show the best performance on image caption generation task. NIC-VA is very different from other compared models, it uses the output of the CNN convolutional layer as image feature map, through the flatten operating, the feature map is changed into 196 vectors. Each vector donates a local feature of the corresponding image. Different local features are imported into LSTM unit at each time-step. At each time-step, the imported word selects local features, therefore, this model is called attention model.

Our model—**FFGS**—shows the best performance on the three datasets. The most important reason is that our **FFGS** is a general model. In other words, the compared models is one of special case of our model. For example, when $\alpha_t = 0$, $g_t = 1$ at every time-step, our model degenerates as LRCN. When only set $g_t = 0$ for all t, our model is changed as NIC-VA. When $\alpha_t = 0$ for all t, $g_t = 1$ at $t = 0$ and $g_t = 1$ for other t, our model degenerates as DeVS. So through training, our FFGS is more robustly than the compared model. The proposed model in this

Table 1. Results of image caption generation on Flickr8K, Flickr30K & MSCOCO

Model	BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR
Flickr8K					
m-RNN	56.5	38.6	25.6	17	-
DeVS	57.9	38.3	14.5	16	16.7
LRVR [1]	-	-	-	14.1	18
Google-NIC	63	41	27	-	-
NIC-VA	67	44.8	29.9	19.5	18.9
Ours	68.2	45.2	31.2	22.5	20.1
Flickr30K					
m-RNN	60	41	28	19	-
DeVS	57.3	36.9	24	15.7	15.3
LRVR [1]	-	-	-	12.6	16.4
Google-NIC	66.3	42.3	27.7	18.3	-
LRCN	58.8	39.1	25.1	16.5	-
NIC-VA	66.7	43.4	28.8	19.1	18.4
Ours	67.9	44.0	29.2	20.9	19.7
MS COCO					
m-RNN	66.8	48.8	34.2	23.9	22.1
DeVS	62.5	45	32.1	23	19.5
LRVR [1]	-	-	-	19	20.4
Google-NIC	66.6	46.1	32.9	24.6	23.7
LRCN	62.8	44.2	30.4	21	-
NIC-VA	68.9	49.2	34.4	24.3	23.9
Ours	70.1	50.3	35.8	25.5	24.1

paper utilizes the global and local image feature which is more comprehensively using the image information.

5 Conclusion

We introduce a feature fusion with gating structure for image caption generation. Both the global image features and the local image features are used to imported into the language model. Through the gating structure, the language model robustly selects the global image features, which solves the argument that the global image feature should be imported into the language model. For the local image features, we used the attention model, which is proved very property

for image caption generation. The proposed method uses the LSTM units as language model. Experimental results show that the proposed image caption generation model is better than all the compared algorithms.

References

1. Chen, X., Lawrence Zitnick, C.: Mind's eye: a recurrent visual representation for image caption generation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2422–2431 (2015)
2. Chung, J., Gulcehre, C., Cho, K., Bengio, Y.: Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555 (2014)
3. Donahue, J., Hendricks, L.A., Rohrbach, M., Venugopalan, S., Guadarrama, S., Saenko, K., Darrell, T.: Long-term recurrent convolutional networks for visual recognition and description. IEEE Trans. Pattern Anal. Mach. Intell. **39**(4), 677–691 (2017)
4. Donahue, J., Anne Hendricks, L., Guadarrama, S., Rohrbach, M., Venugopalan, S., Saenko, K., Darrell, T.: Long-term recurrent convolutional networks for visual recognition and description. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2625–2634 (2015)
5. Fairbank, M., Alonso, E., Prokhorov, D.: An equivalence between adaptive dynamic programming with a critic and backpropagation through time. IEEE Trans. Neural Netw. Learn. Syst. (TNNLS) **24**(12), 2088–2100 (2013)
6. Hodosh, M., Young, P., Hockenmaier, J.: Framing image description as a ranking task: data, models and evaluation metrics. J. Artif. Intell. Res. **47**, 853–899 (2013)
7. Karpathy, A., Fei-Fei, L.: Deep visual-semantic alignments for generating image descriptions. IEEE Trans. Pattern Anal. Mach. Intell. **39**(4), 664–676 (2017)
8. Karpathy, A., Li, F.F.: Deep visual-semantic alignments for generating image descriptions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3128–3137 (2015)
9. Kulkarni, G., Premraj, V., Ordonez, V., Dhar, S., Li, S., Choi, Y., Berg, A.C., Berg, T.L.: BabyTalk: understanding and generating simple image descriptions. IEEE Trans. Pattern Anal. Mach. Intell. **35**(12), 2891–2903 (2013)
10. Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft COCO: common objects in context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8693, pp. 740–755. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10602-1_48
11. Lu, X., Zheng, X., Yuan, Y.: Remote sensing scene classification by unsupervised representation learning. IEEE Trans. Geosci. Remote Sens. **PP**(99), 1–10 (2017)
12. Mao, J., Xu, W., Yang, Y., Wang, J., Huang, Z., Yuille, A.: Deep captioning with multimodal recurrent neural networks (m-RNN). In: International Conference on Learning Representations (ICLR) (2015)
13. Qu, B., Li, X., Tao, D., Lu, X.: Deep semantic understanding of high resolution remote sensing image. In: 2016 International Conference on Computer, Information and Telecommunication Systems (CITS), pp. 1–5, July 2016
14. Rashtchian, C., Young, P., Hodosh, M., Hockenmaier, J.: Collecting image annotations using Amazon's mechanical turk. In: Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk, pp. 139–147. Association for Computational Linguistics (2010)

15. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: International Conference on Learning Representations (ICLR) (2015)
16. Socher, R., Karpathy, A., Le, Q.V., Manning, C.D., Ng, A.Y.: Grounded compositional semantics for finding and describing images with sentences. Trans. Assoc. Comput. Linguist. (TACL) **2**, 207–218 (2014)
17. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1–9 (2015)
18. Vinyals, O., Toshev, A., Bengio, S., Erhan, D.: Show and tell: a neural image caption generator. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3156–3164 (2015)
19. Vinyals, O., Toshev, A., Bengio, S., Erhan, D.: Show and tell: lessons learned from the 2015 MSCOCO image captioning challenge. IEEE Trans. Pattern Anal. Mach. Intell. **39**(4), 652–663 (2017)
20. Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A.C., Salakhutdinov, R., Zemel, R.S., Bengio, Y.: Show, attend and tell: neural image caption generation with visual attention. In: Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6–11 July 2015, pp. 2048–2057 (2015)
21. Young, P., Lai, A., Hodosh, M., Hockenmaier, J.: From image descriptions to visual denotations: new similarity metrics for semantic inference over event descriptions. Trans. Assoc. Comput. Linguist. (TACL) **2**, 67–78 (2014)
22. Zheng, X., Yuan, Y., Lu, X.: Dimensionality reduction by spatial-spectral preservation in selected bands. IEEE Trans. Geosci. Remote Sens. **PP**(99), 1–13 (2017)