Interactive Key-Value Memory-augmented Attention for Image Paragraph Captioning

Chunpu Xu^{1,2*} **Yu Li**^{3} **Chengming Li**^{1} **Xiang Ao**^{4,5} **Min Yang**^{1†} **Jinwen Tian**^{2}
^{1}Shenzhen Key Laboratory for High Performance Data Mining,
Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences
^{2}Huazhong University of Science and Technology
^{3}Shanghai Jiao Tong University
^{4}Institute of Computing Technology, Chinese Academy of Sciences
^{5}University of Chinese Academy of Sciences
cpx@hust.edu.cn, LY0925@sjtu.edu.cn, {cm.li,min.yang}@siat.ac.cn, aoxiang@ict.ac.cn, jwtian@mail.hust.edu.cn

Abstract

Image paragraph captioning (IPC) aims to generate a fine-grained paragraph to describe the visual content of an image. Significant progress has been made by deep neural networks, in which the attention mechanism plays an essential role. However, conventional attention mechanisms tend to ignore the past alignment information, which often results in problems of repetitive captioning and incomplete captioning. In this paper, we propose an Interactive key-value Memory-augmented Attention model for image Paragraph captioning (IMAP) to keep track of the attention history (salient objects coverage information) along with the update-chain of the decoder state and therefore avoid generating repetitive or incomplete image descriptions. In addition, we employ an adaptive attention mechanism to realize adaptive alignment from image regions to caption words, where an image region can be mapped to an arbitrary number of caption words while a caption word can also attend to an arbitrary number of image regions. Extensive experiments on a benchmark dataset (i.e., Stanford) demonstrate the effectiveness of our IMAP model.

1 Introduction

Image captioning has received a significant amount of attention in recent years and is applicable in various scenarios such as virtual assistants, image indexing, and support of the disabled. Significant progress has been made to generate a single sentence to describe an image (Karpathy and Fei-Fei, 2015; Anderson et al., 2018). However, a single sentence has limited descriptive capacity and fails to recapitulate every detail of an image, which largely undermines applications of image captioning in real-world scenarios. One recent alternative to sentence-level captioning is image paragraph captioning with the aim of generating a coherent and fine-grained paragraph (usually 4-6 sentences) to describe an image.

Inspired by the successful use of the encoder-decoder framework employed in neural machine translation (NMT) (Bahdanau et al., 2014), most works on image paragraph captioning employ a convolutional neural network (CNN) as an encoder to obtain fixed-length image representations, and then generates image descriptions with a long short-term memory (LSTM) (Hochreiter and Schmidhuber, 1997) decoder and attention mechanisms. One representative method is to use region-based visual attention to produce a topic vector and then employ language attention to generate caption (Liang et al., 2017).

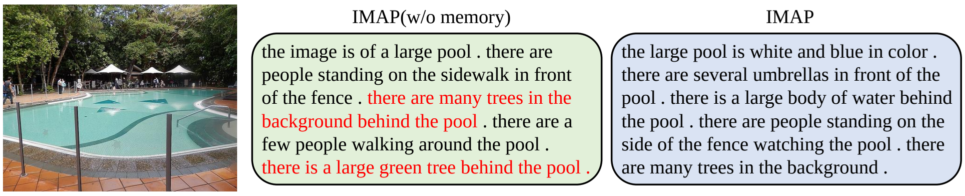
Although conventional attention-based methods have greatly enhanced the performance of image paragraph captioning, there is no mechanism to effectively keep track of the attention history in learning the dynamic alignment between the neural representations of images and that of natural languages. We argue that lacking coverage (history) information might result in two problems in conventional image paragraph captioning: (i) *repetitive captioning* that some image regions are unnecessarily accessed for multiple times and (ii) *incomplete captioning* that some image regions are mistakenly unexplored.

Concretely, the attention at each time step shows which image regions the model should focus on to predict the next target word in the paragraph. However, generating a target word heavily depends on

[*] Work was done when Chunpu Xu was an intern at SIAT.

[†] Min Yang is the corresponding author.

Proceedings of the 28th International Conference on Computational Linguistics, pages 3132–3142
Barcelona, Spain (Online), December 8-13, 2020



IMAP(w/o memory)

the image is of a large pool . there are people standing on the sidewalk in front of the fence . there are many trees in the background behind the pool . there are a few people walking around the pool . there is a large green tree behind the pool .

IMAP

the large pool is white and blue in color . there are several umbrellas in front of the pool . there is a large body of water behind the pool . there are people standing on the side of the fence watching the pool . there are many trees in the background .

Figure 1: Example paragraph captions generated by IMAP and IMAP w/o memory network. The captions generated by IMAP w/o memory contain repeated phrases (in red) and cannot cover all the salient regions of the images.

the relevant parts of the whole image, and an image region is involved in the generation of the whole paragraph. As a result, repetitive captioning and incomplete captioning inevitably happen because of ignoring the coverage of image regions of interest. Figure 1 shows an example from Stanford dataset. The model (i.e., IMAP w/o memory) which is unaware of the coverage information generates repeated sentences to describe the same region of the image ("*there is a large green tree behind the pool*"), while the "*several umbrellas*" and "*a large body of water*" are unexplored.

In this paper, we propose an Interactive key-value Memory-augmented Attention for image Paragraph captioning (IMAP) to alleviate the repetitive captioning and incomplete captioning problems. Our model exploits the recent success of the hierarchical LSTM to generate image paragraph captions (Krause et al., 2017). A sentence LSTM recursively generates sentence topic vectors conditioned on the image features learned by a CNN encoder, and a word RNN is subsequently adopted to decode each topic into output sentence word by word with an attention mechanism to learn the context image representation at each decoding step. Different from conventional attention methods, the IMAP model generates the image context representation that is appropriate for predicting the next target word with iterative memory access operations conducted on a key-memory and a value-memory, inspired by the memory-augmented attention (Meng et al., 2016; Zhang et al., 2017). This mechanism allows the model to track the coverage information typically for each salient object within the image, and therefore avoid repetitive and incomplete captioning. Specifically, we leverage the key-value paired memories to interactively maintain the visual and language features of the input image. The key-memory keeps updated to track the interaction history between the image representation and the decoder by a writing operation, while the value-memory keeps fixed to store the original semantic image features throughout the whole decoding process. In each decoding step, the model learns to address relevant memories based on the key-memory using the "query" state learned from the previous decoder state and previous prediction, and the corresponding values in value-memory are subsequently returned as the image context representation. In addition, we employ an adaptive attention mechanism to realize adaptive alignment from image regions to caption words, where an image region can be mapped to an arbitrary number of caption words while a caption word can also attend to an arbitrary number of image regions.

We summarize our main contributions as follows. (1) We propose an interactive key-value memory-augmented attention to better keep track of attention history and image coverage information, helping the decoder to overcome the repetitive and incomplete captioning problems by automatically distinguishing which parts of the image have been described and which parts are unexplored. (2) We leverage language phrases features with interactive key-value memory-augmented attention to help the model learn better alignment between visual and language features. (3) Experiments on an image paragraph captioning benchmark demonstrate that the proposed method outperforms previous state-of-the-art approaches by a substantial margin, across multiple evaluation metrics.

2 Related Work

Single Sentence Image Captioning Automatic image captioning involves analyzing the visual content of an input image, and generating a textual sentence that verbalizes its most salient aspects (Xu et al., 2015). Inspired by the success of the encoder-decoder framework in neural machine translation, most recent image captioning methods employ the sequence-to-sequence (seq2seq) model to generate image captions (Xu et al., 2015; Anderson et al., 2018; Rennie et al., 2017; Xu et al., 2019; Huang et al., 2019).

For instance, an attention-based encoder-decoder neural network was proposed in (Xu et al., 2015), which learned to dynamically attend to different locations of the images during decoding different words in the captions. Anderson *et al.* (2018) proposed a bottom-up and top-down attention mechanism to enable attention to be computed at the level of objects and salient regions. There were also several recent image captioning studies employing reinforcement learning techniques in the encoder-decoder neural networks. For instance, Rennie *et al.* (2017) presented a self-critical sequence training (SCST) method by considering the optimization of image captioning as a reinforcement learning problem.

Image Paragraph Captioning Since a single sentence has limited descriptive capacity and fails to recapitulate every detail of an image, the task of image paragraph generation has received increasing attention recently, which describes an image with a long, descriptive, and coherent paragraph. Krause *et al.* (2017) was one of the early image paragraph captioning studies, which employed a sentence-level recurrent neural network (RNN) to generate sentence topic vectors, and then applied a word-level RNN to decode each topic vector into a sentence. Subsequently, Liang *et al.* (2017) introduced a recurrent topic-transition generative adversarial network (RTT-GAN) to extend the hierarchical RNN by proposing an adversarial framework between a paragraph generator and two multi-level discriminators (sentence discriminator and topic-transition discriminator). Chatterjee and Schwing (2018) augmented the hierarchical RNN by leveraging coherence vectors to ensure cross-sentence topic smoothness and global topic vectors to summarize the overall information of the image. Wang *et al.* (2019) proposed a convolutional auto-encoding model for image paragraph captioning, which incorporated a convolutional and deconvolutional auto-encoding framework for topic modeling on region-level features of an image.

Different from the aforementioned methods, the IMAP model focuses on alleviating the repetitive and incomplete captioning problems in conventional image paragraph captioning by designing an interactive key-value memory-augmented attention mechanism to track the coverage information typically for each salient object within the image.

3 Our Methodology

Given an image I, image paragraph captioning aims to generate a long paragraph descriptions $Y = \{y_1, y_2, ..., y_N\}$, where N is the number of sentences in the paragraph Y. Each sentence $y_i = \{w_1^{y_i}, w_2^{y_i}, ..., w_T^{y_i}\}$ consists of T words. As illustrated in Figure 2, the proposed IMAP model consists of three parts: (i) an image encoder, which encodes a query image and outputs a set of visual feature vectors and corresponding dense phrases (language features); (ii) a hierarchical decoder, which leverages a sentence-level LSTM to generate sentence topic vectors, and then employs a word-level RNN to decode each topic vector into a sentence word by word; (iii) an interactive key-value memory-augmented attention module, which is used to keep track of the attention history and encourage the decoder to consider the unexplored salient image regions. Next, we will introduce each part of our IMAP model in detail.

3.1 Image Encoder

Following previous works (Johnson et al., 2016; Krause et al., 2017), given an image, we use a dense captioning method, i.e., DenseCap (Johnson et al., 2016) as our image encoder to detect a set of semantic regions and produces the corresponding dense phrases describing the regions in natural language. Formally, we use DenseCap to encode the input image I into M semantic feature vectors, denoted as $V = \{\mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_M\}$. Each semantic feature vector $\mathbf{v}_i$ is a D-dimensional vector ($D = 4096$) corresponding to the features extracted at different locations of the image. Taking the learned visual features vectors V as input, the language model of DenseCap generates a set of dense phrases $S = \{s_1, s_2, ..., s_M\}$, where the phrase s_i corresponds to the semantic visual feature $\mathbf{v}_i$. Each short dense phrase s_i is composed of m words, denoted as $s_i = \{w_1^{s_i}, w_2^{s_i}, ..., w_m^{s_i}\}$.

3.2 Hierarchical Decoder

Inspired by the hierarchical LSTM structure in (Krause et al., 2017), we devise a two-level LSTM-based paragraph generator, which is composed of a sentence-level LSTM (Sent-LSTM) for inter-sentence dependency modeling and two word-level LSTMs (Word-LSTMs) for sentence generation conditioning

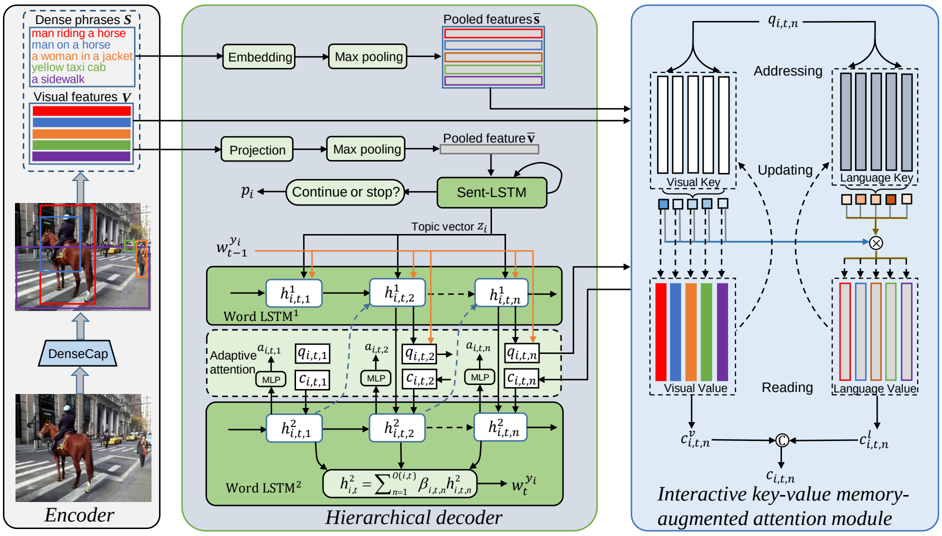


Figure 2: The overview of our IMAP model, which consists of an image encoder, a hierarchical decoder, and an interactive key-value memory-augmented attention module.

on each distilled topic, as illustrated in Figure 2. The sentence LSTM takes as input the image features, and then decides how many sentences to generate in the resulting paragraph and produces an input topic vector for each sentence. Given this topic vector, the word LSTMs generate the sentence word by word. In the decoding step, we also propose an adaptive attention strategy and an interactive key-value memory network to alleviate the repetitive captioning and incomplete captioning problems.

Sentence-level LSTM Similar to (Krause et al., 2017), we aggregate a set of semantic feature vectors V into a single pooled vector $\bar{\mathbf{v}}$ which compactly describes the content of the image. Formally, we compute the pooled vector by projecting each region vector using W and taking a max-pooling operation:

$$\bar{\mathbf{v}} = max_{i=1}^{M}(W\mathbf{v}_i + b) \tag{1}$$

where $W \in \mathbb{R}^{u \times D}$ (u is the dimension of the pooled visual vector) and $b \in \mathbb{R}^u$ are learned parameters, max denotes the max-pooling.

The sentence LSTM decides the number of sentences that should be included in the generated paragraph and produces a u-dimensional topic vector for each of these sentences. It recursively takes the pooled image vector $\bar{\mathbf{v}}$ as input and generates a sequence of hidden states $[\mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_L$, where $\mathbf{h} \in \mathbb{R}^H]$ (H is the dimension of hidden state) and L is the maximum value of the sentence numbers in a paragraph. Each hidden state $\mathbf{h}_i$ is fed into a two-layer fully-connected network to compute the topic vector $\mathbf{z}_i \in \mathbb{R}^u$ for the i-th sentence in the paragraph, which is the input to the word LSTM. In addition, we also use the hidden state $\mathbf{h}_i$ to compute a probability distribution p_i over the two states $\{$CONTINUE=1, STOP=0$\}$ via a linear layer, where the label "CONTINUE" indicates that the decoder should generate next sentence, while "STOP" indicates that the current sentence is the last sentence in a paragraph.

Word-level LSTMs The word-level LSTMs is a two-layer LSTM, which is responsible for generating the words of a sentence. When generating the t-th target word $w_t^{y_i}$ of the i-th sentence, we use the hidden state of the second LSTM layer (LSTM$^{(2)}$) to determine the number of attention steps for the adaptive attention strategy, and derive a weighted average of hidden states of LSTM$^{(2)}$ to generate the target word. Formally, at the n-th attention time step of the decoding time step t, the first word-level LSTM layer (LSTM$^{(1)}$) takes as input the concatenation of the topic vector $\mathbf{z}_i$, the previous output $\mathbf{h}^{(2)}_{i,t,n-1}$ of the second LSTM layer (LSTM$^{(2)}$), and the previous word embedding $e(w_{t-1}^{y_i}) \in \mathbb{R}^E$ (E is the dimension of word embedding):

$$\mathbf{x}^{(1)}_{i,t,n} = [\mathbf{h}^{(2)}_{i,t,n-1}, \mathbf{z}_i, e(w_{t-1}^{y_i})] \tag{2}$$

where $\mathbf{x}^{(1)}_{i,t,n}$ is the input of LSTM$^{(1)}$ at attention time step n of decoding time step t for the i-th sentence, $w_{t-1}^{y_i}$ is the generated word at time step $t-1$. The hidden state of the LSTM$^{(1)}$ at attention time step n

of decoding time step t can be calculated as:

$$\mathbf{h}_{i,t,n}^{(1)} = \text{LSTM}^{(1)}(\mathbf{x}_{i,t,n}^{(1)}, \mathbf{h}_{i,t,n-1}^{(1)}) \tag{3}$$

We use another word-level LSTM (LSTM$^{(2)}$) to produce a caption by generating one word at every time step, which takes as input the concatenation of the output of the LSTM$^{(1)}$ and the context image feature $\mathbf{c}_{i,t,n}$. The hidden state of LSTM$^{(2)}$ at attention time step n of decoding time step t is computed by:

$$\mathbf{h}_{i,t,n}^{(2)} = \text{LSTM}^{(2)}([\mathbf{c}_{i,t,n}, \mathbf{h}_{i,t,n}^{(1)}], \mathbf{h}_{i,t,n-1}^{(2)}) \tag{4}$$

Adaptive Attention Strategy In most previous works (Liang et al., 2017; Chatterjee and Schwing, 2018), each target word attends to only one image region, which is not applicable in practice. In this work, we employ an adaptive attention strategy that allows each word attending to several image regions adaptively. To determine the number of attention steps, a confidence network implemented with a multi-layer perceptron (MLP) is applied to output the probability distribution $a_{i,t,n}$ of each attention step:

$$a_{i,t,n} = \sigma(\text{MLP}(\mathbf{h}_{i,t,n}^{(2)})) \tag{5}$$

The desired attention steps $O(i,t)$ at the t-th decoding step for the i-th sentence is calculated by:

$$O(i,t) = min\{n^{'} : \prod_{n=1}^{n^{'}} (1 - a_{i,t,n}) < \varepsilon\} \tag{6}$$

where ε is a parameter to control the total attention steps, which is set to 10^{-4} in this work.

After finishing all the attention steps, the weighted average of hidden states is calculated as:

$$\mathbf{h}_{i,t}^{(2)} = \sum_{n=1}^{O(i,t)} \beta_{i,t,n} \mathbf{h}_{i,t,n}^{(2)} \tag{7}$$

where

$$\beta_{i,t,n} = \begin{cases} a_{i,t,n} & n = 1 \\ a_{i,t,n} \prod_{n'=1}^{n-1} (1 - a_{i,t,n'}) & n > 1 \end{cases} \tag{8}$$

In conventional attention-based methods, the context vector $\mathbf{c}_{i,t,n}$ is usually calculated as a weighted sum of the whole original image feature vectors V, which ignores the attention history and coverage information of the salient image regions of interest. To make the decoder keep track of previous attention history and attend to the proper image regions at each decoding step, we propose an Interactive Key-value Memory-augmented Attention (IKVMA) to read and update the image feature vectors. We describe the implementation details of IKVMA in Section 3.3. The address operation of IKVMA defined in Eq. (11) and Eq. (12) is used to obtain the visual attention weight, and then the context vector for image features can be computed by the read operation of IKVMA defined in Eq. (13) over the visual key-value memory based on the attention weight, which is denoted as $\mathbf{c}_{i,t,n}^{v}$.

The dense phrases generated by DenseCap provide complementary information for the model to learn better alignment between visual and language features. Thus, we combine the visual features and dense phrases to form the context vector. The pooled language feature vectors $\bar{\mathbf{s}}$ for the input image are computed as:

$$\bar{\mathbf{s}} = [\bar{\mathbf{s}}_1, \bar{\mathbf{s}}_2, ..., \bar{\mathbf{s}}_M], \qquad \bar{\mathbf{s}}_i = \max(\{e(w_j^{s_i})\}_{j=1}^{m_{s_i}}) \tag{9}$$

where $\bar{\mathbf{s}}_i \in \mathbb{R}^E$ is the language feature for the phrase s_i, m_{s_i} is the number of words in the phrase s_i, and $e(w_j^{s_i})$ denotes the word embedding of the word $w_j^{s_i}$. We apply the address operation of IKVMA defined in Eq. (11) and Eq. (12) to compute the language attention weight, and apply the read operation defined in Eq.(13) over a language key-value memory to produce a context vector for dense phrases, which is defined as $\mathbf{c}_{i,t,n}^{l}$. Note that the visual attention weight is reused to make the decoder attend to proper dense phrases when computing the context vector $\mathbf{c}_{i,t,n}^{l}$, via the element-wise product of the visual attention weight and language attention weight.

We concatenate the visual context vector $\mathbf{c}_{i,t,n}^{v}$ and language context vector $\mathbf{c}_{i,t,n}^{l}$ to form the final memory-augmented context vector $\mathbf{c}_{i,t,n} = [\mathbf{c}_{i,t,n}^{v}, \mathbf{c}_{i,t,n}^{l}]$.

Finally, the generation probabilities of the t-th word $w_t^{y_i}$ for the i-th sentence is computed over the entire vocabulary:

$$p(w_t^{y_i}) = \text{softmax}(U_w \mathbf{h}_{i,t}^{(2)} + b_w) \tag{10}$$

where $U_w \in \mathbb{R}^{P \times H}$, $b_w \in \mathbb{R}^P$ are parameters to be learned, P is the number of words in the vocabulary.

3.3 Interactive Key-Value Memory-augmented Attention

The IKVMA module consists of two components: a timely updated key-memory $\mathbf{K} \in \mathbb{R}^{M \times d}$ to keep track of attention history and a fixed value-memory $\mathbf{A} \in \mathbb{R}^{M \times d}$ to store the image features throughout the whole decoding process. Both the key-memory and value-memory consists of M slots, and are initialized with the region feature vectors $V' = \{\mathbf{v}_1^{'}, \mathbf{v}_2^{'}, ..., \mathbf{v}_M^{'}\} \in \mathbb{R}^{M \times d}$, which are obtained by embedding each image feature vector $\mathbf{v}_j \in \mathbb{R}^D$ into a d-dimensional region feature $\mathbf{v}_j^{'}$ through a linear layer. At each decoding step, the j-th slot in key-memory stores the attention status corresponding to the j-th image feature that is updated along with the decoding process, and the j-th slot in value memory stores the representation of the j-th feature vector $\mathbf{v}_j^{'}$.

Key-Memory Addressing At attention time step n of decoding time step t, we get a query vector by taking the concatenation of the hidden state $\mathbf{h}_{i,t,n}^{(1)}$ and the previous word embedding $e(w_{t-1}^{y_i})$ as input:

$$\mathbf{q}_{i,t,n} = \text{LSTM}([\mathbf{h}_{i,t,n}^{(1)}, e(w_{t-1}^{y_i})], \mathbf{q}_{i,t,n-1}) \tag{11}$$

where $\mathbf{q}_{i,t,n}$ is the query vector for the i-th sentence at attention time step n of decoding time step t, which is used to address from the key-memory. Specifically, we compute the attention vector $\alpha_{i,t,n} \in \mathbb{R}^M$ over the visual key-memory $\mathbf{K}_{i,t,n-1}$ as:

$$\alpha_{i,t,n,j} = \frac{\exp(\mu_{i,t,n,j})}{\sum_{j'=1}^{M} \exp(\mu_{i,t,n,j'})}, \quad \mu_{i,t,n,j} = g(\mathbf{q}_{i,t,n}, \mathbf{K}_{i,t,n-1,j}) \tag{12}$$

where g is a two-layer neural network which projects a vector into a scalar value, $\mathbf{K}_{i,t,n-1,j}$ represents the j-th slot of the key-memory at attention time step $n-1$ of decoding time step t for the i-th sentence, $\alpha_{i,t,n,j}$ indicates the weight assigned to the j-th memory slot $\mathbf{K}_{i,t,n-1,j}$.

Value-Memory Reading After obtaining the attention weight $\alpha_{i,t,n}$, the context vector $\mathbf{c}_{i,t,n}$ is computed by the weighted sum of all slots in the value-memory A:

$$\mathbf{c}_{i,t,n} = \sum_{j=1}^{M} \alpha_{i,t,n,j} \mathbf{A}_j \tag{13}$$

where $\mathbf{A}_j$ is the j-th slot in value-memory.

Key-Memory Updating The updating process of the key-memory state includes two operations: ERASE and ADD. The ERASE operation decides the content to be removed from the memory state, which is similar to the forget gate in LSTM. With ERASE operation, the model can avoid exploring the same image location for multiple times, and therefore alleviate the repetitive captioning problem. Formally, the key-memory state after the erase operation is:

$$\tilde{\mathbf{K}}_{i,t,n,j} = \mathbf{K}_{i,t,n-1,j}(1 - \omega_{i,t,n,j}\mathbf{F}_{i,t,n}) \tag{14}$$

where $\mathbf{F}_{i,t,n} = \sigma(W_e, \mathbf{h}_{i,t,n}^{(2)})$, $W_e \in \mathbb{R}^{d \times H}$ is a learnable parameter, σ is the $Sigmoid$ activation function, and $\mathbf{F}_{i,t,n} \in \mathbb{R}^d$. $\omega_{i,t,n,j}$ indicates the weight of the j-th slot of the memory state, which is computed by:

$$\omega_{i,t,n,j} = \frac{\exp(\gamma_{i,t,n,j})}{\sum_{j'=1}^{M} \exp(\gamma_{i,t,n,j'})}, \ \gamma_{i,t,n,j} = g(\mathbf{h}_{i,t,n}^{(2)}, \mathbf{K}_{i,t,n-1,j}) \tag{15}$$

where g is defined in Eq. (12).

The ADD operation decides how much current information (new information) should be added to the visual key-memory state to track the dynamic interaction between the key-memory and the decoder, which is computed as:

$$\mathbf{K}_{i,t,n,j} = \tilde{\mathbf{K}}_{i,t,n,j} + \omega_{i,t,n,j}\tilde{\mathbf{F}}_{i,t,n} \tag{16}$$

where $\tilde{\mathbf{F}}_{i,t,n} = \sigma(W_a, \mathbf{h}_{i,t,n}^{(2)})$, $W_a \in \mathbb{R}^{d \times H}$ is a learnable parameter, and $\tilde{\mathbf{F}}_{i,t,n} \in \mathbb{R}^d$.

3.4 Training Procedure

Our model is trained in an end-to-end manner using the training data (I, Y), where I is an image, and Y is the corresponding human-annotated image description. We assume that Y consists of N sentences and each sentence y_i contains T words. Our overall training loss consists of two cross-entropy losses (the

Method	**BLEU-1**	**BLEU-2**	**BLEU-3**	**BLEU-4**	**METEOR**	**CIDEr**
DenseCap-Concat	33.18	16.92	8.54	4.54	12.66	12.51
Regions-Hierarchical	41.90	24.11	14.23	8.69	15.95	13.52
RTT-GAN	42.06	25.35	14.92	9.21	18.39	20.36
TOMS	43.1	25.8	14.3	8.4	18.6	20.8
CapG-RevG	42.38	25.52	15.15	9.43	**18.62**	20.93
VREN	41.94	24.99	15.01	9.38	17.40	14.71
CAVP	42.01	25.86	15.33	9.26	16.83	21.12
IAP	42.87	26.36	16.07	9.54	16.85	21.81
ICAP	43.38	26.86	16.38	9.72	16.93	22.86
IMAP (Ours)	**44.45**	**27.93**	**17.14**	**10.29**	17.36	**24.07**

Table 1: Comparisons of the proposed IMAP model and the strong baselines on Stanford dataset.

sentence loss and the word loss) and a "attention time loss" used as the time cost penalty for the adaptive attention strategy, which is minimized as follows:

$$J(\theta) = -\lambda_s \sum_{i=1}^{N} \log p_i - \lambda_w \sum_{i=1}^{N} \sum_{t=1}^{T} \log p(w_t^{y_i}) - \lambda_a \sum_{i=1}^{N} \sum_{t=1}^{T} \sum_{n=1}^{O(i,t)} n(1 - a_{i,t,n})) \tag{17}$$

where θ is a set of parameters, p_i is the probability distribution over the states [CONTINUE=1, STOP=0] for the i-th sentence. Note that the target state for the i-th sentence is set to 0 when $i = N$, otherwise 1.

To improve the performance of our model, we apply a policy gradient method (Williams, 1992) to optimize the model after the cross-entropy training by minimizing the negative expected rewards:

$$J(\theta) = -\mathbb{E}_{w_{1:t}^{y_{1:i}} \sim p}[r(w_{1:t}^{y_{1:i}})] \tag{18}$$

where we choose the CIDEr metric as the reward function r.

Following (Rennie et al., 2017), the gradient of the expected rewards can be approximated as:

$$\nabla_\theta J(\theta) \approx -(r(w_{1:t}^{y_{1:i}}) - r(\hat{w}_{1:t}^{y_{1:i}}))\, \nabla_\theta \log p(w_{1:t}^{y_{1:i}}) \tag{19}$$

where $w_{1:t}^{y_{1:i}}$ is a paragraph caption sampled by Monte-Carlo method, and $\hat{w}_{1:t}^{y_{1:i}}$ is a greedy decoding caption paragraph used as the baseline to reduce the variance of the gradient estimate.

4 Experimental Setup

Dataset We conduct the experiments and evaluate our IMAP model on the widely used Stanford image paragraph dataset (Krause et al., 2017), which is the only open source benchmark dataset available for image paragraph captioning. Stanford dataset is collected from MS COCO (Lin et al., 2014) and Visual Genome (Krishna et al., 2017). This dataset consists of 14,575 images for training, 2,487 images for validation, and 2,489 images for testing, each of which has one human-annotated paragraph.

Implementation Details For each image, Faster R-CNN (Ren et al., 2015) initialized with the VGG-16 network (Simonyan and Zisserman, 2014) is applied to detect objects of the image, and top $M = 50$ detected regions are selected as the semantic feature vectors. The size of each feature vector is 4,096, and the word embedding size to 512. We set the maximum number of sentences in each paragraph to $L = 6$, the maximum length of each sentence to 30 via padding operation, and the maximum length of dense phrases to 8. The hidden sizes of both Sent-LSTM and two stacked Word-LSTMs are set to 512. The number of hidden units in the attention layer is 512. We set λ_w, λ_s and λ_a to 1.0, 5.0 and 10^{-4} respectively. The maximum number of attention steps is set to 4. The vocabulary used in the experiment is the same as (Krause et al., 2017). We first pre-train our model with the cross-entropy loss function, and use Adam optimizer (Kingma and Ba, 2014) with an initial learning rate 5×10^{-4} to learn the model. After that, the self-critical training method with CIDEr as the reward is used to further optimize the model. During this stage, the initial learning rate of Adam optimizer is set to 5×10^{-5}.

Baseline Methods In the experiments, we compare the proposed IMAP with the following state-of-the-art methods: DenseCap-Concat (Johnson et al., 2016), Regions-Hierarchical (Krause et al., 2017), RTT-GAN (Liang et al., 2017), TMOS (Mao et al., 2018), CapG-RevG (Chatterjee and Schwing, 2018), VREN (Che et al., 2019), CAVP (Zha et al., 2019), IAP that adopts the Top-Down attention (Anderson et

Method	Cross-entropy						CIDEr-optimization					
	B-1	B-2	B-3	B-4	METEOR	CIDEr	B-1	B-2	B-3	B-4	METEOR	CIDEr
IMAP	42.38	25.87	15.51	9.42	16.56	20.76	44.45	27.93	17.14	10.29	17.36	24.07
w/o memory	40.93	24.32	14.68	8.84	15.71	18.52	42.87	26.36	16.07	9.54	16.85	21.81
w/o language	41.79	25.18	15.23	9.18	16.22	20.04	44.02	27.29	16.75	9.79	17.13	22.98

Table 2: Ablation study on Stanford dataset. Here, B-N is short for BLEU-N.

Model	+2	+1	0	Average score
DenseCap-Concat	0.05	0.56	0.39	0.66
Regions-Hierarchical	0.11	0.63	0.26	0.85
IMAP (w/o memory)	0.20	0.61	0.19	1.01
IMAP	0.28	0.57	0.15	1.13

Table 3: Human evaluation results.

al., 2018) to generate paragraph captions, ICAP that extends the IAP model by using the coverage vector (Tu et al., 2016) to summarize the attention records during the decoding process.

5 Experimental Results

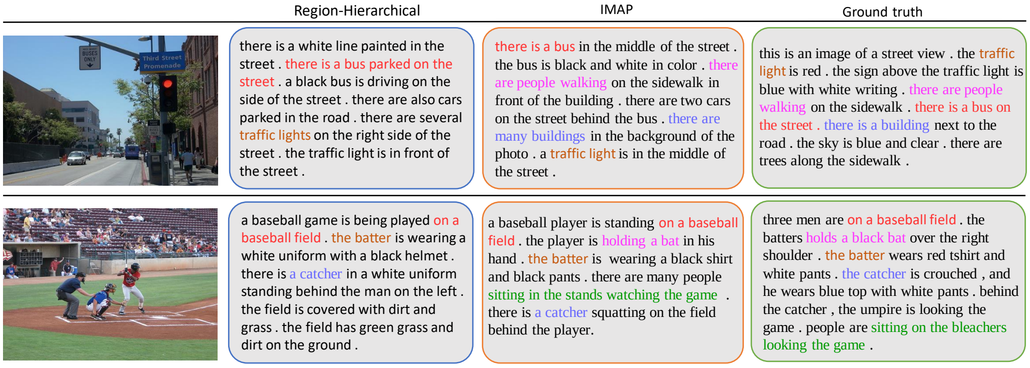
5.1 Automatic Evaluation Results

We quantitatively evaluate our model for across six automatic evaluation metrics that are widely used in previous work (Krause et al., 2017; Liang et al., 2017; Chatterjee and Schwing, 2018), including BLEU-N (N=1,2,3,4) (Papineni et al., 2002), METEOR (Denkowski and Lavie, 2014), CIDEr (Vedantam et al., 2015). These metrics estimate the consistency between the n-gram existence in the produced image descriptions and the ground truth captions.

The experimental results on Stanford dataset are summarized in Table 1. IMAP model achieves significantly better performance than the state-of-the-art competitors on most of the automatic evaluation measures. Concretely, IMAP model successfully yields better scores on all evaluation metrics compared to the Regions-Hierarchical model that utilizes the similar basic hierarchical LSTMs backbone as ours. In addition, IMAP also achieves better scores than ICAP that uses the coverage mechanism to alleviate the repetitive and incomplete captioning problems, which verifies the effectiveness of our interactive memory-augmented attention.

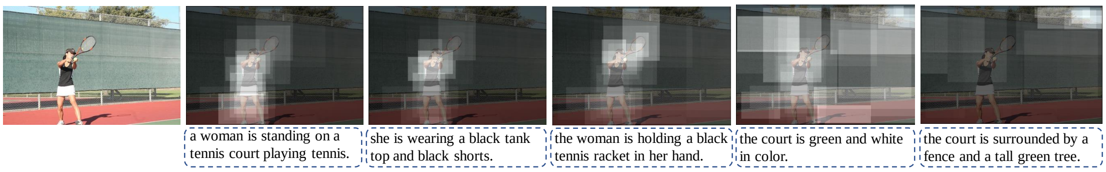
5.2 Ablation Study

To analyze the effect of each component of the IMAP model, we also perform the ablation test of IMAP in terms of discarding the interactive key-value memory network (denoted as w/o memory) and removing the language information (denoted as w/o language). In addition, to investigate the effect of self-critical training method, we demonstrate the experimental results of the models trained with both cross-entropy and CIDEr-optimization methods. The ablation test results are reported in Table 2. We can observe that both the interactive key-value memory-augmented attention and language features contribute greatly to our model. Benefiting from the memory states which keep track of the attention history, the decoder can capture important information and selectively attend to the image regions, especially the unexplored areas, which alleviates the repetitive and incomplete captioning problems. Meanwhile, the language features (dense phrases produced by DenseCap) provide complementary information for the model to learn better alignment between visual and language features, thus the model can generate more precise and fine-grained image descriptions. In addition, we can also observe that the model with policy gradient substantially outperforms the model with cross-entropy by a noticeable margin on all the evaluation metrics. This is because that the policy gradient update is able to bypass the exposure bias and non-differentiable evaluation metrics issue, and maximize long-term reward in paragraph caption generation.



Region-Hierarchical	IMAP	Ground truth
there is a white line painted in the street . there is a bus parked on the street . a black bus is driving on the side of the street . there are also cars parked in the road . there are several traffic lights on the right side of the street . the traffic light is in front of the street .	there is a bus in the middle of the street . the bus is black and white in color . there are people walking on the sidewalk in front of the building . there are two cars on the street behind the bus . there are many buildings in the background of the photo . a traffic light is in the middle of the street .	this is an image of a street view . the traffic light is red . the sign above the traffic light is blue with white writing . there are people walking on the sidewalk . there is a bus on the street . there is a building next to the road . the sky is blue and clear . there are trees along the sidewalk .
a baseball game is being played on a baseball field . the batter is wearing a white uniform with a black helmet . there is a catcher in a white uniform standing behind the man on the left . the field is covered with dirt and grass . the field has green grass and dirt on the ground .	a baseball player is standing on a baseball field . the player is holding a bat in his hand . the batter is wearing a black shirt and black pants . there are many people sitting in the stands watching the game . there is a catcher squatting on the field behind the player.	three men are on a baseball field . the batters holds a black bat over the right shoulder . the batter wears red tshirt and white pants . the catcher is crouched , and he wears blue top with white pants . behind the catcher , the umpire is looking the game . people are sitting on the bleachers looking the game .

Figure 3: Example image captions generated by Region-Hierarchical and IMAP. The words with colors indicate accurate semantic matches between the generated paragraphs and ground truth paragraphs.



a woman is standing on a tennis court playing tennis.

she is wearing a black tank top and black shorts.

the woman is holding a black tennis racket in her hand.

the court is green and white in color.

the court is surrounded by a fence and a tall green tree.

Figure 4: Examples of generated paragraphs with attended image regions. We visualize the mean attention weights on individual pixels for each sentence.The *white* regions indicate the regions where the model roughly attends to when generating the sentences.

5.3 Human Evaluation Results

We also use human evaluation to verify the proposed model. In particular, we randomly selected 200 images from the test set and invited 4 well-educated volunteers to judge the quality of the generated captions of different models. For a generated paragraph caption, a score of +2 indicates the caption is fluent and informative; +1 indicates that the description is fluent but too universal; 0 indicates that the caption is not fluent or contains objects that do not exist in the image. The proportion of each score (0,+1,+2) and the average score are reported for each model. Table 3 demonstrates the results of human evaluation. Consistent with the results of automatic evaluation metrics, the proposed IMAP model can generate more relevant, informative and natural captions than other models.

5.4 Qualitative Results

To evaluate the proposed IMAP model qualitatively, we show some image paragraph captions generated by IMAP and Region-Hierarchical model in Figure 3. IMAP can generate coherent, non-repetitive and comprehensive paragraphs by leveraging the interactive key-value memory-augmented attention to keep track of the image coverage information. On the contrary, Region-Hierarchical (RH) model is prone to generate duplicate phrases within a paragraph. Taking the first case in Figure 3 as an example, the RH model generates repetitive phrases "traffic lights on the right side" and "traffic light is in front" within a paragraph. In addition, it also misses the salient objects "people" and "buildings" in the generated paragraph while IMAP generates the corresponding description " people walking" and "many buildings".

The interactive key-value memory-augmented attention is supposed to keep track of the attention history and the image coverage information. To verify this, in Figure 4, we visualize the attended image regions when generating different sentences in the paragraph. We can observe that IMAP is able to focus on the correct image regions when generating the corresponding sentences. For example, our model can attend to the image object "tennis racket" when generating the sentence "*the woman is holding a black tennis racket in her hand*". The advantage of IMAP comes from keeping track of attention history and image coverage information.

6 Conclusion

In this paper, we proposed an effective interactive key-value memory-augmented attention to alleviate repetitive and incomplete captioning problems in image paragraph captioning, which maintains a timely updated key-memory to track attention history and a fixed value-memory to store the image features during the whole decoding process. To verify the effectiveness of the proposed model, we conducted extensive experiments on the widely used Stanford dataset. The experimental results demonstrated that our model achieved impressive results compared to state-of-the-art image paragraph generation techniques.

Acknowledgement

Min Yang was partially supported by the National Natural Science Foundation of China (NSFC) (No. 61906185), the Natural Science Foundation of Guangdong Province of China (No. 2019A1515011705 and 2018A030313943), the Project of Youth Innovation Promotion Association CAS. Chengming Li was partially supported by Shenzhen Basic Research Foundation (No. JCYJ20180302145607677). Xiang Ao was partially supported by the National Natural Science Foundation of China (No. 61976204, U1811461), the Project of Youth Innovation Promotion Association CAS, Beijing Nova Program (No. Z201100006820062).

References

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

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Moitreya Chatterjee and Alexander G Schwing. 2018. Diverse and coherent paragraph generation from images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 729–744.

Wenbin Che, Xiaopeng Fan, Ruiqin Xiong, and Debin Zhao. 2019. Visual relationship embedding network for image paragraph generation. *IEEE Transactions on Multimedia*.

Michael Denkowski and Alon Lavie. 2014. Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of the ninth workshop on statistical machine translation*, pages 376–380.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Lun Huang, Wenmin Wang, Yaxian Xia, and Jie Chen. 2019. Adaptively aligned image captioning via adaptive attention time. In *Advances in Neural Information Processing Systems*, pages 8942–8951.

Justin Johnson, Andrej Karpathy, and Li Fei-Fei. 2016. Densecap: Fully convolutional localization networks for dense captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4565–4574.

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

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Jonathan Krause, Justin Johnson, Ranjay Krishna, and Li Fei-Fei. 2017. A hierarchical approach for generating descriptive image paragraphs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 317–325.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. 2017. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123(1):32–73.

Xiaodan Liang, Zhiting Hu, Hao Zhang, Chuang Gan, and Eric P Xing. 2017. Recurrent topic-transition gan for visual paragraph generation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3362–3371.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer.

Yuzhao Mao, Chang Zhou, Xiaojie Wang, and Ruifan Li. 2018. Show and tell more: Topic-oriented multi-sentence image captioning. In *IJCAI*, pages 4258–4264.

Fandong Meng, Zhengdong Lu, Hang Li, and Qun Liu. 2016. Interactive attention for neural machine translation. *arXiv preprint arXiv:1610.05011*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99.

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

Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Zhaopeng Tu, Zhengdong Lu, Yang Liu, Xiaohua Liu, and Hang Li. 2016. Modeling coverage for neural machine translation. *arXiv preprint arXiv:1601.04811*.

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

Jing Wang, Yingwei Pan, Ting Yao, Jinhui Tang, and Tao Mei. 2019. Convolutional auto-encoding of sentence topics for image paragraph generation. *IJCAI*.

Ronald J Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pages 2048–2057.

Chunpu Xu, Wei Zhao, Min Yang, Xiang Ao, Wangrong Cheng, and Jinwen Tian. 2019. A unified generation-retrieval framework for image captioning. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pages 2313–2316.

Zheng-Jun Zha, Daqing Liu, Hanwang Zhang, Yongdong Zhang, and Feng Wu. 2019. Context-aware visual policy network for fine-grained image captioning. *IEEE transactions on pattern analysis and machine intelligence*.

Jiani Zhang, Xingjian Shi, Irwin King, and Dit-Yan Yeung. 2017. Dynamic key-value memory networks for knowledge tracing. In *Proceedings of the 26th international conference on World Wide Web*, pages 765–774. International World Wide Web Conferences Steering Committee.