

# Effective Multimodal Encoding for Image Paragraph Captioning

Thanh-Son Nguyen<sup>1</sup> and Basura Fernando<sup>1,2</sup>

<sup>1</sup>*Agency for Science, Technology and Research (A\*STAR), Singapore*

<sup>2</sup>*Centre for Frontier AI Research, A\*STAR, Singapore*

**Abstract**—In this paper, we present a regularization-based image paragraph generation method. We propose a novel multimodal encoding generator (MEG) to generate effective multimodal encoding that captures not only an individual sentence but also visual and paragraph-sequential information. By utilizing the encoding generated by MEG, we regularize a paragraph generation model that allows us to improve the results of the captioning model in all the evaluation metrics. With the support of the proposed MEG model for regularization, our paragraph generation model obtains state-of-the-art results on the Stanford paragraph dataset once further optimized with reinforcement learning. Moreover, we perform extensive empirical analysis on the capabilities of MEG encoding. A qualitative visualization based on t-distributed stochastic neighbor embedding (t-SNE) illustrates that sentence encoding generated by MEG captures some level of semantic information. We also demonstrate that the MEG encoding captures meaningful textual and visual information by performing multimodal sentence retrieval tasks and image instance retrieval given a paragraph query.

**Index Terms**—Multimodal encoding generation, image paragraph captioning, text generation, autoencoder.

## I. INTRODUCTION

DESCRIBING images using natural language has become an important problem in Artificial Intelligence. While image captioning (e.g., [1], [2]) focuses on generating a simple description, image paragraph generation [3] aims to describe an image in more details. Therefore, image paragraph generation is more challenging than image captioning and in particular, a paragraph generation model should be able to generate multiple coherent and consistent sentences describing the details of an image. Image paragraph generation is important for image understanding [4], image retrieval [5], instance retrieval, and the development of assisting technologies for visually impaired people [6], [7]. The primary societal impact of this paper is in the domain of generating descriptions of the environment, e.g., generating scene descriptions to be played to vision impaired people so that they can be aware of the surrounding environment. This type of technology may also assist elderly people to navigate in urban environments as well. Furthermore, the outcomes of this research are useful to the research line of developing industrial applications such as maintenance report generation from visual inspections and fault diagnosis report generation from visual inspection (Figure 2). The most common approach to image paragraph generation is to use hierarchical recurrent neural network conditioned on images [3], [8]. Several methods use sentence

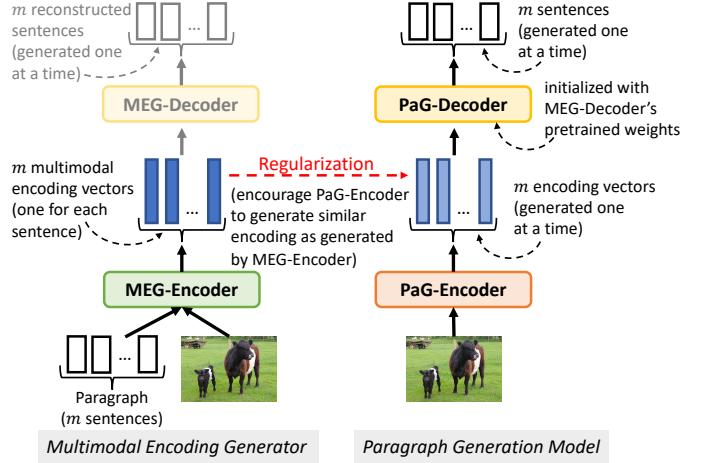


Fig. 1: Training the paragraph generation model (PaG) using the pretrained multimodal encoding generator (MEG). The multimodal encoding generated by MEG is used to *regularize* PaG by constraining the encoding generated by PaG-Encoder to be similar to the corresponding multimodal encoding generated by MEG-Encoder. Moreover, PaG-Decoder is initialized with MEG-Decoder’s pretrained weights to utilize the reconstruction capability of MEG.

encoding [3] or topic modelling [8] to obtain coherent and consistent sentences.

Writing a good paragraph is not an easy task, even for humans. An informative paragraph consists of a key sentence describing the scene of the image, followed by supporting sentences. To articulate an accurate depiction of the image, one should keep this structure to make sure the paragraph is focused on the main idea. Furthermore, paragraphs should build a scene and continue a good narrative. Obviously, the image paragraph depends on the visual content, the information the writer wants to convey, creativity of the writer, and writing skills. Inspired by the observation, we propose to first train a multimodal encoding generator (MEG) model to capture the semantics, writing skills, coherence, and structure of human-generated image paragraphs. We train MEG such that the multimodal encoding generated by this model captures the high-level information useful to *reconstruct* human-generated sentences (i.e., paragraph). Then, we transfer the knowledge captured by MEG’s encoder to a paragraph generation model (PaG) using regularization. Specifically, during training, PaG



Fig. 2: Examples of the applications of paragraph generation from images. Paragraph generation can be used for assistive technologies for visual impaired people, to generate visual inspection reports and to generate maintenance reports by visual inspection. (The sources of the left image and the right image are wikipedia.org and www.qualitymag.com, respectively)

is encouraged to generate encoding that is similar to the corresponding multimodal encoding generated by MEG. This allows the captioning model to generate more accurate paragraphs. This high-level idea is shown in Figure 1.

MEG learns to encode a sentence based on the structure of the paragraph and the corresponding image. MEG consists of an encoder and a decoder as shown in Figure 1. The encoder takes an image and the ground-truth paragraph as input and generates an encoding vector for each sentence. The decoder takes each encoding vector and vision features to reconstruct the corresponding input sentence, one by one, thus reconstructing the input paragraph. For the decoder to generate correct reconstruction, the encoder should generate informative and representative encoding. Our MEG’s encoder and decoder are designed to make use of both visual and textual information to reconstruct a paragraph. By doing so, we aim to transfer human writing skills to our MEG model. MEG encapsulates this capability in the latent representations (i.e., multimodal encoding). In Section V, we investigate the textual and visual information encoded in the multimodal encoding generated by MEG. For example, Figure 7 shows the sentence encoding obtained by MEG is semantically informative, and visually and textually consistent. Our objective is to transfer this information into our image paragraph generation model.

Multimodal encoding generated by MEG is used to guide the training process of PaG (Figure 1). PaG maintains the state of a paragraph using a long short-term memory (LSTM) and at each time step, PaG’s encoder generates a sentence encoding which is passed to PaG’s decoder to generate a sentence for the given image. For each sentence, we constrain the sentence encoding generated by PaG to be close to the corresponding multimodal encoding generated by MEG. We regularize the PaG model by adding a term to the training loss that minimizes the distance between sentence encoding generated by PaG and the corresponding sentence encoding generated by the pretrained MEG. Moreover, PaG and MEG share the same decoder’s structure and thus, we initialize PaG’s decoder with

the pretrained MEG’s decoder. This is to maximize the benefit of transferring information from MEG to PaG. In other words, for PaG to “inherit” MEG’s reconstruction capability.

Our main contributions are as follows. First, we propose a multimodal encoding generator, MEG, which learns an embedding space that can capture image (visual) and paragraph (textual) information. The learnt embedding space (or multimodal encoding) supports reconstructing input paragraphs with outstanding accuracy. Secondly, we show how to employ MEG for improving a paragraph generation model for the image paragraph captioning task. Experimental results show that MEG helps improve the captioning model for both before and after fine-tuning using reinforcement learning. Moreover, the best setting of our model achieves new state-of-the-art performance for BLEU scores with comparable METEOR and CIDEr scores on the Stanford paragraph dataset. We also present a thorough evaluation of the learnt multimodal encoding and show that the encoding generated by MEG contains some level of visual and textual semantics. This explains the improvements of the captioning model when being regularized by MEG during training.

## II. RELATED WORK

One of the pioneering works on image paragraph generation is proposed in [3]. The model consists of a visual feature detector and a hierarchical paragraph generator. Pooled visual features are passed through the first recurrent neural network (RNN) also known as the sentence RNN. The sentence RNN generates a representation vector of the sentence and then this vector is decoded by the word RNN to generate words of the sentence. Our model also follows a slightly similar approach. However, the sentence encoding generated by our paragraph generation model, PaG, is regularized by a pre-trained multimodal encoding generator, MEG. MEG encodes a sentence explicitly conditioning on both paragraph and image information. Therefore, the multimodal encoding generated by MEG is used to transfer the structural knowledge of a paragraph to the captioning model. The method proposed in [3] has a simpler training procedure compared to ours, but our model is more regularized.

Liang et al. [8] use a Sequence GAN-based architecture to improve paragraph generation. In [8], the sentence generator generates sentences while the sentence and the topic transmission discriminators measure the plausibility and smoothness of semantic transition with preceding sentences. This method also uses sentence encoding which is, however, used as a topic vector to determine the topic of the next sentence and to determine the end of a paragraph. Our sentence encoding is regularized by a pre-trained recurrent sentence autoencoder that exploits the visual information, sentence, and paragraph structure explicitly. While it may seem the topic vectors in [8] should have semantic meaning to function properly, in reality the topic vectors of [8] may not have a semantic meaning. Our latent code obtained by MEG does not necessarily need to possess semantic meaning; it just needs to capture information in the input sentence and relevant information from the image.

Hierarchical supervision consisting of hierarchical rewards and values at both sentence and word levels is proposed

in [10]. In a similar spirit to our method, this hierarchical reward method provides dense supervision for the paragraph generator. In our case, sentence level supervision and regularization is provided by the explicit sentence encoding while the word level supervision is provided by the traditional cross-entropy loss. Mao et al. [11] use Latent Dirichlet Allocation (LDA) [12] topics of sentences to improve the paragraph generation by maximizing the likelihood of joint sentence and topic distribution of a given image. Similar to us, LDA topics could act as a regularizer. However, one difficulty is that LDA topics have highly overlapping words, and some images might not be well presented by the topic distribution and therefore regularization may not be as effective as in our proposed method.

Reinforcement learning (RL) is also widely adopted to train [13] or fine-tune [10], [14] paragraph generation models. Self-critical learning, a form of RL, is a highly effective technique employed by many image captioning and paragraph generation methods [15], [16]. Although not as common, convolutional neural network (CNN) is also adopted in visual paragraph generation models [17]. Yang et al. [18] propose the scene graph auto-encoder (SGAE) method to incorporate inductive biases in language into the image captioning model. It uses a scene graph autoencoder which reconstructs an input sentence via a scene graph generated by a fixed off-the-shelf scene graph language parser. The authors generate a scene graph from the image and obtain an embedding from the scene graph. Then the authors use knowledge distilled from the language-based scene graph auto-encoder to improve image-based scene graph embedding for captioning tasks. Our work is similar to [18], however, we do not rely on the intermediate step of generating the scene-graphs which itself is a particularly challenging problem.

There are other methods exploring additional visual cues to improve paragraph generation, e.g., Wang et al. [19] exploit the depth maps obtained by external models to further enrich the visual representation. In a similar approach, Che et al. [20] use visual relationship detection to improve paragraph generation. The model in [20] detects regions which may contain important visual objects and then predicts their relationships. Paragraphs are produced based on object regions which have valid relationships with the others. Yang et al. [21] use image scene graph to improve paragraph generation. In principle, whichever additional visual cue can potentially enrich the visual representation and could improve the quality of generated paragraphs. In our work, we focus more on the paragraph generation technique rather than the visual representation. However, we believe the works of [20], [19], [21] are complementary to ours.

More recently, Gupta et al. [22] propose a method to obtain a paragraph level text embedding for paragraph generation. This method tries to eradicate the inconsistency between the parallel extraction of visual features and sequential text supervision and so called "Text Embedding Bank" (TEB), learns a fixed-length feature representation from a variable-length paragraph. TEB acts as a form of global and coherent deep supervision to regularize visual feature extraction in the image encoder. It is also used as a distributed memory to

provide features of the whole paragraph to the language model, which alleviates the long-term dependency problem. A video paragraph generation method is also presented [23]. Special attention has been devoted to generating diverse paragraphs. There are attempts to incorporate various levels of external knowledge and logical reasoning in visual question answering [4]. These contributions are orthogonal to the regularization technique presented in our work.

Our main idea is conceptually similar to model distillation ([24], [25], [26]) where a teacher network teaches the student network to behave similar to the teacher. However, in our case we use the teacher network outputs to regularize the paragraph generation model.

In general, our method is also related to image captioning [27], [28], [29], video captioning [30], [31], and other vision and language tasks [32], [33]. Yao et al. [27] proposed to inject attribute information along with the image features to the RNN model to boost captioning performance. In [31], tags from the image are used to improve the captioning performance. They use the probability of each tag to learn the parameters of the LSTM model. Our approach is different from these ideas as we rely on obtaining a multimodal representation for both image and the sentences and use these representations to regularize the paragraph generation.

### III. MULTIMODAL ENCODING GENERATOR

A captioning model creates a paragraph for an image by generating sentence by sentence. Each sentence is generated based on a latent vector, called *sentence encoding* whose space is traditionally learnt during training the captioning model. In this paper, we propose to first learn the encoding space by separately training a *multimodal encoding generator* (MEG) for the task of paragraph reconstruction. Then, we use the learnt encoding space to regularize the *paragraph generation model* (PaG) during training. Specifically, given an image having feature  $V$ , PaG creates a paragraph caption by generating sentences one by one. To generate the  $k^{th}$  sentence, it first computes sentence encoding  $\hat{z}_k$  using its encoder.

$$\hat{z}_k = G(V) \quad (1)$$

where  $G$  is the function representing for the computations in PaG's encoder. During training, PaG is guided to generate sentence encoding to be similar or close to the corresponding multimodal encoding generated by the pre-trained MEG, i.e., to minimize the distance  $L_D(z_k, \hat{z}_k)$ , where  $L_D$  is a distance function (e.g.,  $L_2$  distance),  $z_k$  is the multimodal encoding for sentence  $k^{th}$  in the corresponding ground-truth paragraph  $P$ .  $z_k$  is computed using MEG's encoder as follows.

$$z_k = F(P, V) \quad (2)$$

where  $F$  is the function representing for the computations in MEG's encoder. By doing this, we "transfer" the knowledge learnt by MEG to the captioning model, PaG. Moreover, as MEG's decoder and PaG's decoder share the same structure, PaG's decoder is initialized with MEG's decoder's pretrained weights to utilize the reconstruction capability of MEG.

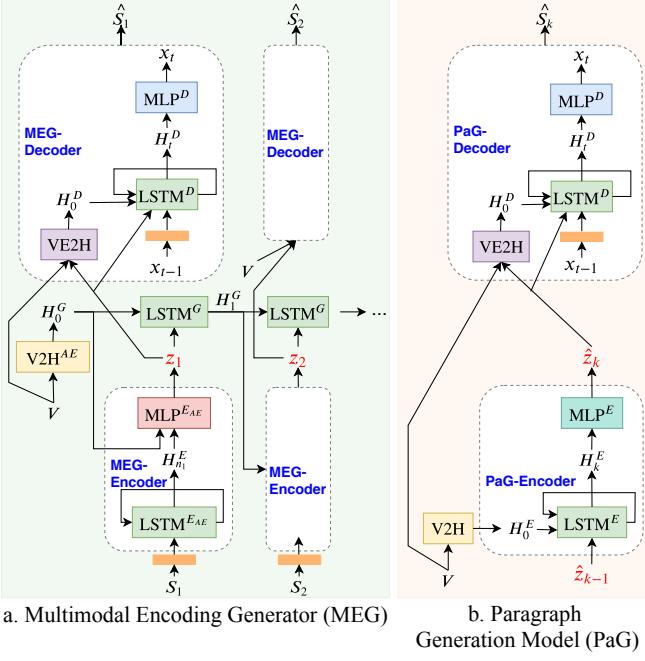


Fig. 3: Our proposed models. MEG incorporates visual and paragraph-sequential information to generate multimodal sentence encoding which in turn, is used to regularize the captioning model, PaG.

We now present our proposed multimodal encoding generator (MEG) which aims to learn an effective multimodal encoding space and guide the training process of the captioning model. MEG’s objective is to generate multimodal encoding for each sentence that contains visual, textual and sequential information. Let us denote the vision feature set of an input image by  $V$  (e.g., features extracted from Faster RCNN regions [34]). The corresponding ground-truth paragraph  $P = \{S_1, \dots, S_m\}$  contains  $m$  sentences, where  $S_k = \{x_1, \dots, x_{n_k}\}$  ( $k = [1 \dots m]$ ) denotes for the sentence at position  $k$  that contains  $n_k$  words. Word  $x_i \in \mathbb{R}^{d_C}$  is a  $d_C$ -dimensional one-hot vector and  $d_C$  is the vocabulary size.  $z_k$  denotes for the multimodal encoding for  $S_k$ . We first train the MEG model to generate multimodal encoding  $z_k$  that takes into account not only the sentence  $S_k$ , but also the corresponding paragraph  $P$  and the image.

MEG aims to generate representative encoding of a given sentence conditioning on the image and the paragraph. This encoding can reconstruct the sentence within the paragraph. Traditional autoencoder only consider a single input sentence when generating encoding. However, for the encoding to be useful for the image paragraph generation task, it needs to take into account (1) the other sentences in the same paragraph and (2) the image that it is describing for. In other words, the encoding for the same sentence should be different when the sentence is placed in different paragraphs or for different images. Therefore, we use a “global” LSTM ( $LSTM^G$ ), to capture sequential information of the whole paragraph by taking encoding as input. The hidden state  $H_k^G$  of the  $LSTM^G$  captures the visual and paragraph level information. An in-

put sentence is first passed to a LSTM of MEG’s encoder ( $LSTM^{E_{AE}}$ ). Then the last hidden state of  $LSTM^{E_{AE}}$ , together with the previous hidden state of  $LSTM^G$  are used to compute the sentence encoding. Vision features are used to compute the initial hidden state of  $LSTM^G$ . Figure 3a illustrates our proposed MEG model.

#### A. The network structure of MEG-Encoder

MEG’s encoder consists of a LSTM and a multilayer perceptron (MLP). An input sentence  $S_k$  is first encoded using the encoder’s LSTM ( $LSTM^{E_{AE}}$ ) as follows.

$$H_{n_k}^E = f(S_k W_E, H_0^E; \Phi_{E_{AE}}) \quad (3)$$

where  $H_0^E \in \mathbb{R}^{d_{E_{AE}}}$  and  $H_{n_k}^E \in \mathbb{R}^{d_{E_{AE}}}$  are the initial and the last  $d_{E_{AE}}$ -dimensional hidden states, respectively;  $f$  is LSTM operations applying on the whole sequence and returns the last hidden state;  $\Phi_{E_{AE}}$  is trainable parameters of the LSTM;  $W_E \in \mathbb{R}^{d_C \times d_w}$  is the word embedding matrix and  $d_w$  is word embedding size. The sentence encoding is then computed based on the last hidden state of  $LSTM^{E_{AE}}$  and the previous hidden state of  $LSTM^G$ :

$$z_k = \tanh(W_1^z (H_{n_k}^E \oplus H_{k-1}^G)) \quad (4)$$

where  $z_k \in \mathbb{R}^{d_z}$  is the  $d_z$ -dimensional sentence encoding for the input sentence  $S_k$ ;  $W_1^z \in \mathbb{R}^{(d_{E_{AE}}+d_G) \times d_z}$  is trainable parameters;  $\oplus$  is the concatenation operation;  $H_{k-1}^G \in \mathbb{R}^{d_G}$  is the  $d_G$ -dimensional hidden state at time  $k-1$  of  $LSTM^G$  that is computed by:

$$H_{k-1}^G = \begin{cases} f(z_{k-1}, H_{k-2}^G; \Phi_G), & \text{if } k \geq 2 \\ g_v(V), & \text{if } k = 1 \end{cases} \quad (5)$$

where  $f$  is LSTM operations,  $\Phi_G$  are trainable parameters of  $LSTM^G$ . The initial hidden state ( $H_0^G$ ) is computed using  $g_v(V)$ , a transformation on self-attended vision features:

$$g_v(V) = \tanh(W_{g_v} V_{att_S}) \quad (6)$$

where  $W_{g_v} \in \mathbb{R}^{d_V \times d_G}$ ;  $V_{att_S} \in \mathbb{R}^{d_V}$  is the  $d_V$ -dimensional self-attended vision features.  $V_{att_S}$  is computed as the weighted average of  $V$  using the attention weights  $\mathbf{a}_{V_S}$  which is computed following [35].

$$\mathbf{a}_{V_S} = \text{softmax}(W_1^{a_S} \tanh(W_2^{a_S} V)) \quad (7)$$

where  $V \in \mathbb{R}^{d_r \times d_V}$  is vision features containing  $d_r$   $d_V$ -dimensional vectors;  $W_2^{a_S} \in \mathbb{R}^{d_r \times d_V}$  and  $W_1^{a_S} \in \mathbb{R}^{d_r \times 1}$ .

#### B. The network structure of MEG-Decoder

The encoding is then passed to the decoder to reconstruct the input sentence. The decoder uses  $LSTM^D$  to generate a word at each time step. The initial hidden state is computed as a transformation of encoding-based visual attention:

$$H_0^D = \tanh(W_{H_D} V_{att_E}) \quad (8)$$

where  $H_0^D$  is the  $d_D$ -dimensional hidden state of  $LSTM^D$ ;  $W_{H_D} \in \mathbb{R}^{d_V \times d_D}$  and  $V_{att_E}$  is the encoding-based visual attention which is the weighted average over vision features

using the attention weights  $\mathbf{a}_{V_E}$  which computed as follows [35]:

$$\mathbf{a}_{V_E} = \text{softmax}(W_1^{a_E} \tanh(W_2^{a_E}(V \oplus z_k))) \quad (9)$$

where  $W_2^{a_E} \in \mathbb{R}^{d_r \times (d_V + d_z)}$ ,  $W_1^{a_E} \in \mathbb{R}^{d_r \times 1}$ . The hidden state of  $\text{LSTM}^D$  at time  $t$  is computed as:

$$H_t^D = f(z_k \oplus (x_{t-1} W_E), H_{t-1}^D; \Phi_D) \quad (10)$$

where  $x_{t-1} \in \mathbb{R}^{d_C}$  is the one-hot vector for the word at time  $t-1$ ,  $W_E$  is the word embedding matrix, and  $\Phi_D$  is the  $\text{LSTM}^D$  parameters. The output distribution is then computed as follows.

$$p(x_t) = \text{softmax}(W_1^o \text{relu}(W_2^o H_t^D)) \quad (11)$$

where  $W_2^o \in \mathbb{R}^{d_D \times d_D}$ ,  $W_1^o \in \mathbb{R}^{d_D \times d_C}$  and  $d_C$  is the vocabulary size. Word  $x_t$  is then sampled using this output distribution. The decoder stops when “stop-of-sentence” token is generated.

### C. Training MEG

MEG is trained by minimizing the reconstruction loss which is the cross entropy loss between the input sentence  $S_k$  and the generated sentence  $\hat{S}_k = \{\hat{x}_1, \dots, \hat{x}_{n_k}\}$ :

$$\arg \min_{\Omega} \frac{1}{n_k} \sum_{t=1}^{n_k} \sum_{i=1}^{d_C} -x_t^i \log p(\hat{x}_t^i) \quad (12)$$

where  $\Omega = \{\Phi_{EAE}, \Phi_G, \Phi_D, W_1^z, W_{g_v}, W_1^{a_S}, W_2^{a_S}, W_{H_D}, W_1^{a_E}, W_2^{a_E}, W_1^o, W_2^o\}$  are trainable parameters and  $i = [1 \dots d_C]$  is the index of word in the vocabulary.

Similar to traditional *autoencoders*, our MEG is inherently designed in a way that it is not possible to simply copy the input to the output since the decoder does not have direct access to the input sentence. Moreover, both  $\text{LSTM}^G$  and  $\text{LSTM}^D$  in MEG are conditioned on both visual and textual information, it cannot learn an identity mapping of the input text, but rather have to learn to generate meaningful multimodal sentence encoding  $z$  that is simultaneously representative for the input sentence, the input image, and the sequential information of the input paragraph. The experimental results in Section V-D show that MEG is able to reconstruct unseen paragraphs accurately demonstrating its generalization capacity.

## IV. PARAGRAPH GENERATION MODEL

PaG adopts the hierarchical recurrent neural network [36] that consists of paragraph-level network (PaG-Encoder) and sentence-level network (PaG-Decoder) to handle the sequential information of paragraph level and sentence level, respectively. Figure 3b shows our captioning model, PaG.

### A. The network structure of PaG-Encoder and PaG-Decoder

**PaG-Encoder.** At each time step, PaG-Encoder first generates an encoding which is then used by PaG-Decoder to generate a sentence. PaG-Encoder consists of a LSTM ( $\text{LSTM}^E$ ) and a multilayer perceptron ( $\text{MLP}^E$ ). The hidden state of

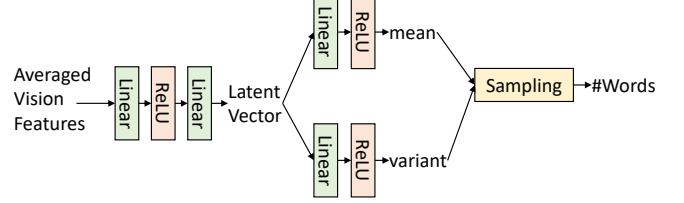


Fig. 4: NModel takes as input the averaged vision features, predicts the mean and variant which are then used to sample the number of words needed to describe the image.

$\text{LSTM}^E$  is initialized by a transformation of the self-attended vision features similar to Equation 6:

$$H_0^E = \tanh(W_{H_E} V_{att_S}) \quad (13)$$

where  $H_0^E \in \mathbb{R}^{d_E}$  is the initial  $d_E$ -dimensional hidden state of  $\text{LSTM}^E$ ;  $W_{H_E} \in \mathbb{R}^{d_V \times d_E}$ ;  $V_{att_S}$  is the weighted average of vision features  $V$  using attention weights computed as in Equation 7. The hidden state at time  $k$  is computed as follows.

$$H_k^E = f(\hat{z}_{k-1}, H_{k-1}^E; \Phi_E) \quad (14)$$

where  $\hat{z}_{k-1}$  is the previously generated sentence encoding and  $\Phi_E$  is LSTM parameters. The encoding for sentence at time  $k$  is then computed by:

$$\hat{z}_k = \tanh(W_2^z H_k^E) \quad (15)$$

where  $W_2^z \in \mathbb{R}^{d_E \times d_z}$ .

**PaG-Decoder.** To make use of the pretrained decoder in MEG, PaG-Decoder has the same structure as in MEG-Decoder (Section III-B), and is initialized with the pretrained MEG-Decoder’s parameters. PaG-Decoder follows the procedures in MEG-Decoder to generate a sentence given sentence encoding and vision features. The model stops when the generated content reaches a predetermined length.

### B. Training the paragraph generation model

In addition to ground-truth paragraphs, we use the multimodal encoding vectors generated by the pretrained MEG as training signals. Specifically, PaG is trained by minimizing the cross entropy loss ( $L_{XE}$ ) and encoding distances ( $L_D$ ):

$$\arg \min_{\Psi} (\alpha L_{XE} + (1 - \alpha) L_D) \quad (16)$$

where  $L_{XE}$  is computed using Equation 12;  $\alpha$  is the cross-entropy loss weight;  $\Psi = \{\Phi_E, \Phi_D, W_{H_E}, W_2^z, W_1^{a_S}, W_2^{a_S}, W_E, W_{H_D}, W_1^{a_E}, W_2^{a_E}, W_1^o, W_2^o\}$  denotes for the trainable parameters of the model (noted that the parameters for the decoder use the same notations as in the decoder of MEG);  $L_D$  is the distance between encoding generated by PaG and encoding generated by pretrained MEG.

### C. Estimating the number of words to be generated

To identify when a sentence ends, generation models typically generate *<EOS>* (end-of-sentence) token. Similarly, to identify when a paragraph ends, we tried to train a paragraph generation model which generates *<EOP>* (end-of-paragraph)

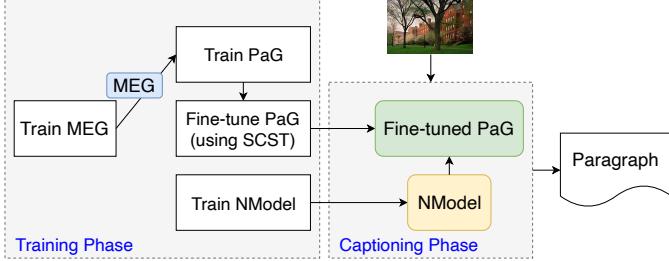


Fig. 5: The two-phase workflow showing the steps for training the models and performing the paragraph captioning task.

token, but it did not work. One reason is due to the much longer content of paragraph compared to a sentence makes it more difficult to learn generating the stop signal. Alternatively, we can predict the length of the paragraph to be generated. The length can be the number of sentences or the number of words. One issue with predicting the number of sentences is that we do not control the actual length of the paragraph since sentences can have different number of words. Therefore, we propose to predict the number of words a paragraph should have to describe an image. This is similar to the number-of-word constraint when writing an essay. We train a separate model, namely *NModel* (Figure 4), which takes averaged vision features as input and predicts the mean and variance of the distribution. The number of words is then sampled and passed to the captioning model which will keep generating new sentences until the number of words is reached. *NModel* is trained by optimizing the  $L_1$  loss between the prediction and the ground-truth number of words in a paragraph. *NModel* is trained once and used in all different settings of the generation model (unless otherwise specified).

Figure 5 shows the workflow of our proposed method that includes 2 phases: *training phase* and *captioning phase*. In the training phase, we first train the Multimodal Encoding Generator (MEG) which is then used for regularization during training the Paragraph Generation model (PaG). PaG is then fine-tuned based on *self-critical sequence training* (SCST) [15]. A separate model, namely *NModel*, is trained to predict the number of sentences needed to describe a given image. The fine-tuned PaG and *NModel* are used in the captioning phase, in which, given an image, the model generates a paragraph description.

## V. EXPERIMENTS

### A. Dataset and Experimental Settings

**Dataset.** We evaluate our proposed method using the Stanford image paragraph dataset [3] which contains 19,561 images and each image has a human-generated paragraph. To be comparable with other methods, we follow the data splitting as in [3], i.e., there are 14,575 images in train set, 2487 images in validation set and 2489 images in test set. Performances on the test set are reported.

**Baselines.** We compare with the state-of-the-art methods evaluated using the Stanford image paragraph dataset that are categorized into four main groups: sentence/topic encoding-based, autoencoder, reinforcement learning, and alternative

vision-based approaches. **Sentence/topic encoding-based** approach includes *Regions-Hierarchical* [3] (combining region features and hierarchical RNN to generate paragraphs), *RTT-GAN* (Semi + Fully) [8] (a recurrent topic-transition generative adversarial network containing a sentence generator and two discriminators for assessing sentence plausibility and topic coherence), and *TOMS* [11] (a topic-oriented multi-sentence captioning model which uses a pretrained Latent Dirichlet allocation to guide the model generating topic embedding and generates a sentence for each topic). **Autoencoder** approach includes *CAPG-VAE* [37] (a variational autoencoder modelling diversity and coherence of generated sentences in a paragraph) and *CAE-LSTM* [38] (a convolutional auto-encoding which learns to generate “topics” to be used for generating sentences). **Reinforcement learning** approach includes *SCST-w-RP* [16] (adopted top-down model from [39] and combined self-critical sequence training with repetition penalty), *DHPV* [10] (a densely supervised hierarchical policy-value network which contains a sentence-level and word-level value modules to evaluate the values of preceding sentences and words for generating a paragraph), *CRL* (2-beam) [13] (a curiosity-driven reinforcement learning framework which incorporate both intrinsic and extrinsic rewards using pure reinforcement learning), and *CAVP* [14] (a context-aware visual policy network which considers visual attentions as context and decides whether the context is used for generating current word/sentence given the current visual attention). We report the best setting of CAVP, i.e. the fine-tuned model with BLEU optimization). The other methods are grouped into **alternative vision-based** approaches including *DAM* [19] (a depth-aware attention model which leverages depth estimation to infer object-object spatial relations), *VRD* [20] (a visual relationship detection-based model that also generates sentences based on objects and spatial relations), *Twin-ParaCNN* [17] (a CNN-based model with adversarial twin net training scheme), and *HSGED (SLL)* [21] (a hierarchical scene graph encoder-decoder which uses image scene graph as the script to integrate semantic knowledge into the model). The best settings of the baselines are used and the results are obtained from the corresponding papers.

**Fine-tuning.** We adopt *reinforcement learning* (RL), specifically *self-critical sequence training* (SCST) [15] to fine-tune our captioning models. The objective is to maximize the expected rewards (i.e., evaluation score for a single metric or a set of metrics), or minimize the negative expected rewards. The rewards are normalized by a predefined baseline function to reduce the variance of policy gradient. In SCST, the output obtained by test-time inference algorithm is utilized as the baseline. Our experimental results showed that optimizing a combination of evaluation metrics achieves better results than that of optimizing for a single metric. Specifically, after performing a grid search, the best set of weights are 1.0, 0.8, and 0.01 for BLEU-4, METEOR, and CIDEr, respectively.

**Settings.** We adopt the bottom-up attention object detector (BU) [39] to obtain vision features for images. Given an image, BU generates a feature vector  $V \in \mathbb{R}^{d_r \times d_V}$  where  $d_V = 2048$  and  $d_r$  varies depending on the number of regions of the image. The word embedding matrix ( $W_E$ ) is initial-

Abbreviation	Meaning
MEG	Multimodal Encoding Generator
PaG	Paragraph Generation Model
PaG-NoReg	PaG not regularized by MEG
PaG-NoReg-SCST	PaG-NoReg fine-tuned with SCST
PaG-MEG	PaG regularized by MEG
PaG-MEG-SCST	PaG-MEG fine-tuned with SCST
PaG-MEG-FD	PaG-MEG with freezing the decoder
PaG-MEG-FD-SCST	PaG-MEG-SCST with freezing the decoder

TABLE I: Abbreviations of our models with different settings

PaG Variants	B_1	B_2	B_3	B_4	MET	CID
PaG-NoReg	36.17	19.13	10.94	6.35	13.45	11.52
PaG-MEG-FD	27.33	11.07	4.10	1.38	9.78	4.78
PaG-MEG	41.78	24.44	14.35	8.25	15.51	18.19
PaG-NoReg-SCST	40.99	23.65	14.12	8.39	14.77	18.07
PaG-MEG-FD-SCST	38.92	21.66	12.16	6.72	14.21	11.47
PaG-MEG-SCST	<b>46.96</b>	<b>29.57</b>	<b>18.61</b>	<b>11.51</b>	<b>18.24</b>	<b>29.43</b>

TABLE II: Evaluating the use of MEG in training paragraph captioning. PaG-MEG and PaG-NoReg denote for the captioning models with and without using MEG, respectively. Suffix “SCST” indicates the settings using self-critical sequence training for fine-tuning. PaG-MEG-FD and PaG-MEG-FD-SCST denote for the captioning models which freeze their decoders when training and fine-tuning, respectively. The results show that MEG consistently helps improve the captioning model’s performance regarding all the metrics and in both with and without fine-tuning. Moreover, fine-tuning the paragraph captioning model’s decoder helps improve the performance.

ized by the pretrained GloVe embeddings [40],  $d_w = 300$ . During training the multimodal encoding generator, the word embedding matrix  $W_E$  is freezed and only be fine-tuned during training the paragraph generation model. We keep 2000 most-frequent words which have GloVe embeddings as the dictionary (i.e.,  $d_C = 2000$ ). Our LSTMs are uni-directional and are implemented using the formulations in [41]. We use  $L_2$  distance as the encoding loss  $L_D$  (Equation 16). We train the paragraph length predictor, NModel, to minimize the  $L_1$  loss between the predicted number of words and the ground-truth number of words in a paragraph. The latent vector’s dimension and the batch size are 128 and 1000, respectively. The best NModel is chosen based on the  $L_1$  loss. By default, the same pretrained NModel is used to determine the length of the paragraph to be generated for all of our paragraph generation models. Adam optimizer [42] is utilized to train our models.

**Evaluation metrics.** We report the results for the widely used metrics in text generation, i.e., BLEU-{1,2,3,4} [43], METEOR [44], and CIDEr [45] (denoted as B\_{1, 2, 3, 4}, MET, and CID, respectively). We use MS COCO evaluator [46] to compute the scores. Results using these metrics are reported in percentage (i.e. multiplied by 100%).

### B. Effects of the multimodal encoding generator on the paragraph generation model

To evaluate the values of the multimodal encoding generator (MEG), for the task of paragraph captioning, we compare different settings of the paragraph generation model (PaG)

with the options of regularization using MEG-encoding and finetuning using SCST. Table I shows the different settings and abbreviations used.

Table II shows the comparison results. The first two result lines in Table II show that using MEG helps PaG gain relatively huge improvements regarding all the evaluation metrics. With MEG, the generation model gained more than 30% improvement for BLEU-3, BLEU-4 and CIDEr.

We further investigate the effects of MEG on PaG when finetuning with reinforcement learning. We examine whether applying reinforcement learning “overwrites” the effect of using MEG. In other words, whether MEG consistently helps improving the performance even with the use of reinforcement learning, particularly SCST. As shown in Table II (PaG-NoReg-SCST and PaG-MEG-SCST), SCST improves the results of the captioning model (PaG-NoReg-SCST), and interestingly, with the additional use of MEG, the performance significantly improves further (PaG-MEG-SCST). The improvement is even stronger compared to before finetuning. The percentage improvements are 32%, 37%, and 63% for BLEU-3, BLEU-4, and CIDEr, respectively.

These results show that the multimodal encoding generator consistently helps improve the paragraph captioning model regardless of finetuning using SCST for all the evaluation metrics. The benefits of using MEG are not “overwritten” or affected by the fine-tuning technique. In other words, MEG acts as a *complementary component* for improving the quality of paragraph captioning models.

**Should we fine-tune PaG’s decoder?** A hypothesis is that since PaG’s encoder is trained to generate sentence encoding close to the ground-truth encoding (i.e., encoding generated by MEG’s encoder), PaG will work the best when using the pretrained MEG’s decoder as its decoder. To examine this hypothesis, we train the paragraph captioning when PaG’s decoder is initialized with the pretrained MEG’s decoder and freezed during training and fine-tuning PaG (all the other training factors and procedure are the same as for obtaining PaG-MEG and PaG-MEG-SCST). The results are reported in Table II where PaG-MEG-FD and PaG-MEG-FD-SCST denote for the captioning models using the setting of freezing the decoder when training and fine-tuning PaG, respectively. Paragraph captioning models when freezing the decoders achieve significantly lower results than those with fine-tuning the decoders (e.g., BLEU-4 score for PaG-MEG-FD is 1.38, compared to 8.25 BLEU-4 score for PaG-MEG; and BLEU-4 score for PaG-MEG-FD-SCST is 6.72, compared to 11.51 BLEU-4 score for PaG-MEG-SCST). These results are practically expected due to the tasks’ requirements. Instead of generating encoding for the current input sentence (as in MEG), PaG’s encoder is trained to generate the encoding of the next sentence. Therefore, generating the exact ground-truth encoding is very challenging (if not impossible). Allowing PaG to fine-tune its decoder helps the model learn better since we provide the flexibility for the decoder to adjust with the learning capability of the PaG’s encoder. It is worth mentioning that with fine-tuning the PaG’s decoder, paragraph captioning training obtains the best performance much quicker than when training with freezing PaG’s decoder. The best epochs obtained

Approach	Method	BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR	CIDEr
Sentence/Topic Encoding-based	Regions-Hierarchical [3]	41.90	24.11	14.23	8.69	15.95	13.52
	RTT-GAN (Semi+Fully) [8]	42.06	25.35	14.92	9.21	18.39	20.36
	TOMS [11]	43.10	25.80	14.30	8.40	18.60	20.80
Autoencoder	CAPG-VAE [37] CAE-LSTM [38]	42.38 -	25.52 -	15.15 -	9.43 9.67	18.62 <b>18.82</b>	20.93 25.15
Reinforcement Learning	SCST-w-RP [16]	43.54	27.44	17.33	10.58	17.86	30.63
	DHPV [10]	43.35	26.73	16.92	10.99	17.02	22.47
	CRL (2-beam) [13]	43.12	27.03	16.72	9.95	17.42	31.47
Alternative Vision-based Approaches	CAVP [14]	42.01	25.86	15.33	9.26	16.83	21.10
	DAM [19]	35.00	20.20	11.70	6.60	13.90	17.30
	VRD [20]	41.74	24.94	14.94	9.34	17.32	14.55
Our generation model	Twin-ParaCNN [17]	43.30	25.80	15.60	9.50	17.20	20.60
	HSGED(SLL) [21]	44.51	28.69	18.28	11.26	18.33	<b>36.02</b>
PaG-MEG-SCST	<b>46.96</b>	<b>29.57</b>	<b>18.61</b>	<b>11.51</b>		18.24	29.43

TABLE III: Comparing the best setting of our proposed model with state-of-the-art methods for image paragraph captioning. The baseline methods are grouped by their main approach categories. Our proposed model, PaG-MEG-SCST, outperforms all the baseline methods regarding BLEU scores and the results are comparable for METEOR and CIDEr.

for training PaG (no SCST) with fine-tuning PaG’s decoder and with freezing PaG’s decoder are epoch 29 and epoch 103, respectively. These results show the advantages of fine-tuning PaG’s decoder when using the guidance provided by MEG. In the next section, we use our best setting (i.e., PaG-MEG-SCST) to compare with state-of-the-art methods.

### C. Comparison with the baseline models

We compare our proposed model with the state-of-the-art methods for the image paragraph captioning task. We use our best setting in which the paragraph generation model is regularized by our proposed multimodal encoding generator and fine-tuned using SCST (PaG-MEG-SCST). Table III shows the comparison results. Our model outperforms all the baseline methods regarding all the BLEU scores, and are comparable for METEOR and CIDEr. The BLEU scores are not correlated to CIDEr and METEOR, thus it is entirely possible that they show different trends. This happens especially in paragraph evaluation compared to image captioning where the paragraphs are much longer than single-sentence captions (e.g., MS COCO dataset [47]). In paragraph generation, the METEOR scores would not change much unless the model is significantly better than all other compared methods. This can be seen in our results in METEOR for all methods in TABLE III. However, CIDEr scores are very sensitive to the generated paragraph length. If our model knows the length of the paragraph, then our model obtains a CIDEr score of 82.48 as shown later in the experiments in TABLE VII. Additionally, Figure 8 illustrates the effects of predicting correctly the paragraphs’ lengths on the evaluation metrics. More detailed discussion is in Section V-E. Noted that among the baselines, none of the methods achieves the best performance for all the metrics.

The closest approach groups to our model are sentence/topic encoding-based and reinforcement learning. As shown in Table III, our model performs better than the methods in these groups for five out of six evaluation metrics. Interestingly, our model outperforms other reinforcement learning-based methods by

Image	Ground-truth Paragraph (input sentences)	Reconstructed Sentences
	the food is filled with vegetables.	the food is filled with vegetables.
	there is and broccoli and all other of vegetables.	there is and broccoli and all other of vegetables.
	the food is bright and with color.	the food is bright and with color.
	the food is still in the pan getting cooked.	the food is still in the pie still hot.
	there is a train on the train tracks.	there is a train on the train tracks.
	the train is black with white numbers on the front of it.	the train is black with five white numbers on the wall.
	in the distance you can see trees with green leads and a black metal fence.	in the distance you can see trees with green and a fenced clock up cement.
	a pizza sits on a yellow plate.	a pizza sits on a yellow plate.
	there are cut up tomatoes on top of the pizza.	there are cut up tomatoes on top of the pizza.
	there is spinach on top of the pizza.	there is spinach on top of the pizza.

TABLE IV: Sentences (paragraph) reconstructed by the pre-trained MEG model. MEG is able to encode and reconstruct almost perfectly the input sentences. (Only words included in the dictionary are displayed)

a considerable margin (except for CIDEr). The experimental results show that the proposed model is effective for the task of image paragraph captioning. In the next sections, we thoroughly analyze the multimodal encoding generated by MEG, especially to see if the multimodal encoding contains any levels of semantic information.

### D. Analyzing the multimodal encoding generated by MEG

1) *Paragraph Reconstruction:* The multimodal encoding generator is trained for the task of paragraph reconstruction. The model is trained for 200 epochs and the last epoch is selected. The performance of the pre-trained MEG for the

No	Query	Top 5 (query excluded)
1	there is a white chair behind the cat.	there is a white wall behind the bed. there is a blue pillow in front of the cat. there is a black tile wall behind the toilet. there is a white sink in front of the toilet. there is a woman in a blue shirt beside the toilet.
2	there is spinach on top of the pizza.	there is green leaves on top of the pizza. cheese is on top of the pizza. there is cheese and lettuce on top of the pizza. there is mustard on top of the hot dog. there is food on top of the table.
3	cars are parked on the street near the sidewalk.	there is a truck parked on the street next to the sidewalk. there are a couple of cars parked on the street near the sidewalk. many cars are parked next to the sidewalk. there is a gray car on the street beside the bus. there are three cars parked on the street behind the horses.
4	an airplane is flying in the sky.	a plane is flying in the sky. a plane is flying in the blue sky. a plane is flying in the air. a kite is in the sky. the man is jumping in the air.
5	the umbrella is a light pink color.	the towel is a light pink color. the sand is a light beige color. the cloth is a red color. the sky is a light blue. the floor is a light tan color.

TABLE V: Top similar sentences returned for each *query* based on cosine similarity scores computed using sentence encoding. Query and the closest sentences convey similar concepts. (Only words appeared in the dictionary are shown.)

task of paragraph reconstruction is as follows (in percentage): **BLEU-1:** 84.09, **BLEU-2:** 78.34, **BLEU-3:** 74.47, **BLEU-4:** 71.44, **METEOR:** 45.81 and **CIDEr:** 657.63<sup>1</sup>. The high results show the capability of reconstructing input sentences. In other words, the encoding generated by MEG is useful for reconstructing the desired sentences, thus can be used as additional information for training the paragraph generation model.

Table IV demonstrates MEG’s capability of reconstructing input paragraphs. By considering both vision and textual information as input, MEG is able to reconstruct the input sentences. In the first image, the first three sentences are perfectly reconstructed. Similarly for the sentences in the other two images, the reconstructed sentences are exactly the same or slightly different from the input sentences (but having the same meaning). This explains the very high evaluation scores of the multimodal encoding generator. During training the paragraph generation model, one of the objective is to generate encoding that is close to the multimodal encoding generated by the pre-trained MEG. In theory, if the generation model (PaG) can generate the exact encoding as generated by MEG, PaG can perform as good as MEG’s reconstruction since PaG’s decoder is initialized with MEG’s decoder’s weights.

2) *Does multimodal encoding contain any textual semantics?:* To evaluate the *textual* information encoded in the multimodal encoding generated by MEG, we conducted an

experiment in which the objective is to search for *similar sentences* based on multimodal encoding. We first compute the multimodal encoding for each sentence in the test set by passing a paragraph and the corresponding image as input to the pre-trained MEG. The task now becomes, given a sentence (with its multimodal encoding) as query, return a list of the most relevant sentences. The relevance between two sentences is computed as the cosine similarity between their corresponding multimodal encoding vectors. Table V shows five queries and the five most relevant sentences for each query. The top relevant sentences convey a similar concept with the query. For example, query 1 and its relevant sentences describe relative position between objects, i.e., “*behind*”. Interestingly, the sentences are not matched by the word “*behind*” only, but rather seems to be able to match by the meaning, that is why the top relevant sentences also contain “*in front of*” which relatively conveys similar semantic. Likewise for query 3, we have “*near*”, “*next to*”, and “*beside*”. Query 2 and query 4 are also about the concepts of spatial and positional. Whereas, query 5 is about object’s property (i.e., color). The results show that the multimodal encoding generated by MEG encodes some level of textual semantic. Noted that the multimodal encoding is generated based on both textual and visual information. One potential-practical application of this sentence retrieval task is to find similar news articles, where both text and images are present. The multimodal encoding generated by MEG will be useful since it incorporates both textual and visual information.

3) *Does multimodal encoding contain cross-modality (visual-textual) information?:* Our hypothesis is that sentence encoding generated by MEG encodes both *visual* and *textual* information. To examine the interaction between the cross-modality information within sentence encoding, we perform an *image instance retrieval* (IIR) task which takes a paragraph as query and returns the image of interest. Specifically, given a set of images  $I = \{I_1, \dots, I_T\}$  and a paragraph  $P$  as query ( $P$  is the ground-truth paragraph of an image in  $I$ ), the goal is to retrieve the corresponding image based on MEG and PaG models. Images are ranked based on their *relevance* scores regarding the input query  $P$ . We compare two different methods of computing relevance scores: (1) *text-based IIR* which bases on paragraph generated by the captioning model, PaG, and (2) *encoding-based IIR* which bases on multimodal encoding.

*Text-based IIR.* We first use the captioning model, PaG, to generate paragraphs for all the images in  $I$ . The relevance score of an image is measured as the cosine similarity based on TF-IDF (Term Frequency - Inverse Document Frequency) between its generated paragraph and the query  $P$ :

$$R_T(I_k, P) = \text{sim}(P_k^F, P^F) \quad (17)$$

where  $R_T(I_k, P)$  is the text-based relevance score between image  $I_k$  and the query paragraph  $P$ ,  $P_k$  is the generated paragraph for image  $I_k$ ,  $P^F$  and  $P_k^F$  are the TF-IDF vectors representing for  $P$  and  $P_k$ , respectively. The *sim* function is defined as the cosine similarity. We compare the results when using PaG-NoReg and PaG-MEG for generating paragraphs.

<sup>1</sup>CIDEr-D is used and it can be greater than 100 (%)

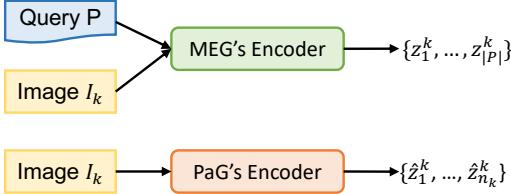


Fig. 6: Encoding-based Image Instance Retrieval

$K$	Text-based IIR PaG-NoReg	Text-based IIR PaG-MEG	Encoding-based IIR
1	2.49	<b>4.38</b>	2.61
5	6.67	<b>9.84</b>	8.52
10	10.08	<b>14.91</b>	14.79
15	12.70	18.44	<b>19.61</b>
20	15.31	21.90	<b>23.70</b>
50	24.19	34.79	<b>39.05</b>
100	33.83	45.84	<b>53.52</b>

TABLE VI: Precision at  $K$  for the task of image instance retrieval based on paragraph as the query. All the paragraph generation models used in this experiment are not fine-tuned.

To focus on the effect of MEG only, both models are not fine-tuned with SCST.

*Encoding-based IIR.* For each image in  $I$ , we compute two versions of encoding by using encoders from MEG and PaG as illustrated in Figure 6. Given an input query  $P$  and an image  $I_k$ , we first use the MEG’s encoder to generate the multimodal encoding for each sentence in  $P$ , i.e.,  $\{z_1^k, \dots, z_{|P|}^k\}$ . Then, we use the captioning model to generate a paragraph for image  $I_k$ , but only keep the sentence encoding of the generated sentences, i.e.,  $\{\hat{z}_1^k, \dots, \hat{z}_{n_k}^k\}$ . Notice that the number of generated sentences  $n_k$  can be different from the number of sentences in  $P$ . The relevance score for image  $I_k$  is then computed as the average of the maximum similarity score between the encoding of each sentence in  $P$  with all the encoding vectors of the generated sentences for  $I_k$  as follows.

$$R_E(I_k, P) = \frac{1}{|P|} \sum_{i=1}^{|P|} \max\{sim(z_i^k, \hat{z}_j^k) \mid j = 1 \dots n_k\} \quad (18)$$

where  $R_E(I_k, P)$  is the encoding-based relevance score between image  $I_k$  and the query paragraph  $P$ ,  $|P|$  is the number of sentences in  $P$ ,  $sim(z_i^k, \hat{z}_j^k)$  is the cosine similarity between two sentence encoding vectors  $z_i^k$  and  $\hat{z}_j^k$ .

Table VI shows the precision at  $K$  scores (i.e., whether the correct image is shown in the top  $K$  results) for text-based IIR methods and encoding-based IIR method. As mentioned, we use two variants of text-based IIR: 1) paragraph generation model without using MEG (PaG-NoReg) and 2) captioning model with MEG as regularization (PaG-MEG). The results showed that text-based IIR PaG-MEG performs much better than text-based IIR PaG-NoReg thanks to the information provided by MEG during training the generation model. The encoding-based IIR method performs comparably with text-based IIR PaG-MEG for top 5 and 10, and interestingly performs the best from top 15 and above. Although the text-based IIR methods have the advantage of having the decoders

to generate the sentences before matching with the query, the decoders seems to ignore some useful information in the multimodal encoding, thus explains encoding-based IIR performs better with high value of top  $K$  (i.e., from top 15).

To qualitatively examine the returned results, we visualize the top 3 images returned by each method for 4 queries in Table VIII. For the first query, all of the methods are able to retrieve images of buses, but only encoding-based method is able to return an image with all the details of bus, decorations, people standing, and tall buildings. In the second query, text-based PaG-NoReg does not return any images of “kite” in the top 3. Whereas for both text-based PaG-MEG and encoding-based, the top 2 images are about people flying kite. Although fail with its top 3, encoding-based method’s top 2 images have, interestingly, the details of “the sky is full of white and gray clouds”. The third query is an example where both text-based methods fail to return relevant images while encoding-based method’s results are relevant and the top 2 images are about “snowboarder jumping”. In the last image, text-based methods are not doing well, there is only one image relevant to airplane in text-based PaG-MEG’s top 2. Encoding-based method is able to retrieve all relevant images of airplane flying.

4) *Visualizing the multimodal encoding generated by MEG:* Similar to word embedding, the multimodal encoding aims to represent for a sentence and, in the context of MEG, the corresponding image. The question is that whether the multimodal encoding generated by MEG contains any semantic of the textual and/or visual information. We qualitatively investigate this research question by checking “what makes two multimodal encoding vectors close to each other”. Figure 7 visualizes the multimodal encoding vectors generated by MEG for all the sentences in the test set, based on t-distributed Stochastic Neighbor Embedding (t-SNE) [48]. It shows that close vectors tend to represent for similar topic or concept, i.e., (two) animals standing, train, people gathering, street and vehicles, people and weather, and playing tennis. Moreover, the figure showcases MEG’s capability of simultaneously encoding visual and textual information. E.g., one group of images contains ‘trains’ and the textual description obtained from the sentence encoding reveals similar concepts (e.g train tracks). Trains are visually observable, yet train tracks are not. Perhaps the term “tracks” is obtained by the language modality. It is interesting to see that our sentence encoding is both visually and textually enriched. The qualitative results show that the sentence encoding generated by MEG contains some level of semantic information. This also explains for the improvement achieved when using MEG’s generated encoding as regularizer for training the captioning model.

#### E. The effect of paragraph length on the evaluation metrics

Since we use a separate model (NModel) to predict the number of words to generate, we examine the effect of paragraph length on the evaluation metrics. We compare the performance of the generation model when the generated paragraphs are longer or shorter than the ground-truth paragraphs by a certain number of words. We vary the number of words difference from -50 to 50. When the number of words difference is 0,

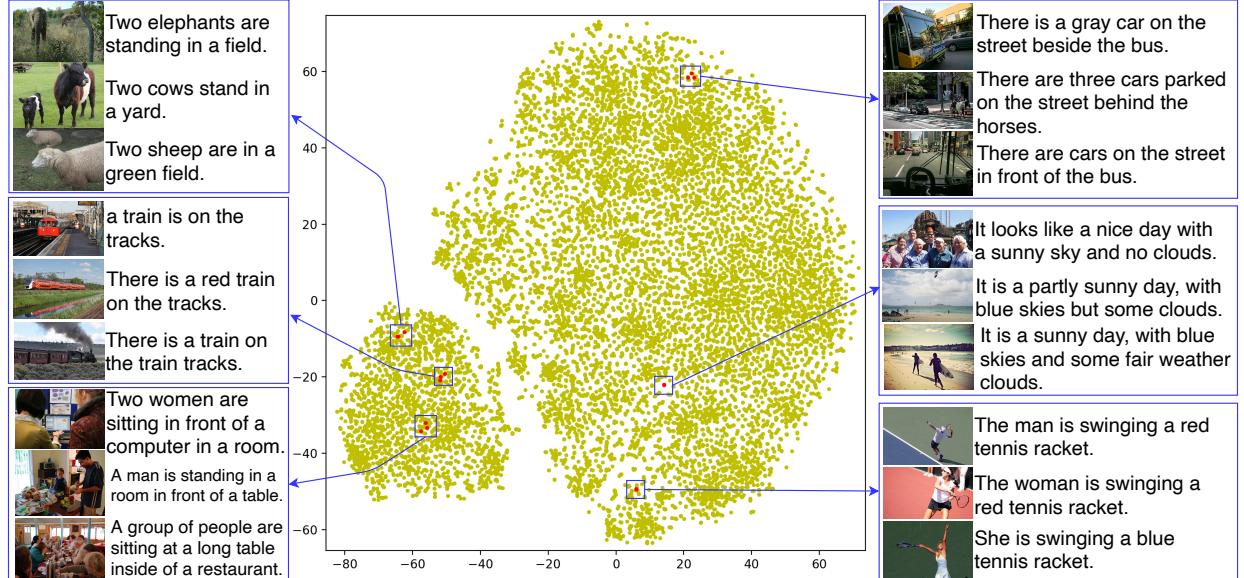


Fig. 7: Visualization of sentence encoding vectors (generated by MEG) of all the sentences in the test set by applying t-distributed Stochastic Neighbor Embedding (t-SNE). The visualization illustrates that sentence encoding contains *semantic* information that closer sentence encoding vectors in the space represent for sentences having similar *topic* or *concept*.

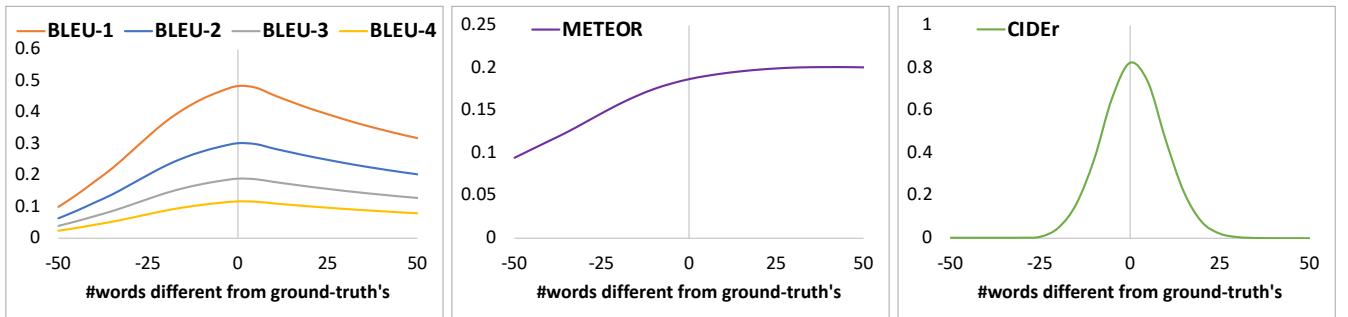


Fig. 8: The effect of generated paragraph's length on the evaluation metrics. Generating shorter paragraphs reduces the performance for all the metrics. While affecting BLEU scores and CIDEr, generating longer paragraphs does not affect METEOR. CIDEr is very sensitive to the length of generated paragraphs.

the generation model tries to generate a paragraph having the same length as the ground-truth's. We use the best setting of our generation model for this experiment (i.e., PaG-MEG-SCST). Figure 8 shows the scores of all the evaluation metrics when varying the number of words difference. Generating paragraphs having different lengths from the ground-truth paragraphs affects all the metrics, but not in the same way. METEOR is only affected when generating shorter paragraphs and it is even slightly increased for longer paragraphs. Whereas, the BLEUs and CIDEr are influenced by the length difference for both longer and shorter generated paragraphs. They reach the best value when generated paragraphs have the same length as the ground-truth's. CIDEr is extremely sensitive to the length difference as it decreases close to zero when the absolute value of the length difference greater or equal to 25. We also evaluate the performance of the NModel using mean L1 error during both training and testing. Our NModel obtains mean L1 error of 17.5 during training and mean L1 error of 18.5 during testing. These L1 distances (i.e.,

average number of words different from ground-truth's) and our model performance are associated with the findings from our experiment regarding the effect of generated paragraph's length on the evaluation metrics reported in Figure 8.

To evaluate how the prediction from NModel affects the final results, we compare the performances of paragraph generation models when using ground-truth number of words and prediction from NModel. We use the paragraph generation model trained with MEG as regularizer, i.e., PaG-MEG, and test the model before and after finetuning with SCST. Table VII shows the comparison results. Except for CIDEr, the scores using NModel are comparable to those using the ground-truth number of words for both before and after finetuning. This is understandable since as shown in Figure 8, CIDEr is extremely sensitive to the length difference. Nevertheless, the CIDEr score when using NModel is comparable with the state-of-the-art baselines (Table III). This proves NModel's capability of assisting the generation model. NModel is only trained once and used as an additional

SCST	#Words	B_1	B_2	B_3	B_4	MET	CID
No	NModel	41.78	24.44	14.35	8.25	15.51	18.19
	GT	42.14	24.42	14.26	8.16	16.29	36.85
Yes	NModel	46.96	29.57	18.61	11.51	18.24	29.43
	GT	48.45	30.25	18.96	11.73	18.69	82.48

TABLE VII: Evaluating the use of NModel in predicting the number of words (#Words). All results are obtained with PaG-MEG using #Words from ground-truth (GT) and NModel. The performance using NModel is comparable to that obtained with GT #Words for both with and without fine-tuning (SCST) (only except for CIDEr).

component for the generation models.

#### F. Case study: qualitatively evaluate the effect of using MEG in training paragraph generation model

In this section, we compare the paragraphs generated by the fine-tuned paragraph generation models in which one model was not trained with MEG (i.e., PaG-NoReg-SCST) and one model was trained with MEG (i.e., PaG-MEG-SCST). Table IX shows the ground-truth paragraphs and generated paragraphs for seven different images. For the generated paragraphs, we highlight in different colors for the information that is **correct**, **wrong**, **unsure** (if the information is right or wrong), or **repeated**.

As shown in Table IX, without using information from MEG, the paragraph model tends to generate repeated sentences with less details. Whereas, when training with MEG, the paragraph model provides more detailed description about the input image without generating duplicated sentences. For example, in image 1, without MEG, the generation model can only mention the “window on the wall next to the bed”, but with MEG, the generation model can describe many details such as “a bed is in a room”, “window on the wall next to the bed”, “white pillows on top of the bed”, “the bed is red and white”, “there is a small window on the side of the bed”.

MEG is helpful for training the generation model to detect more accurate information about objects and actions. For example, PaG-NoReg-SCST detected “a bunch of airplanes are parked on the runway” for image 2, but in fact it is about “two airplanes are flying in the sky” as generated by the generation model trained with MEG. Another example is in image 5 where PaG-NoReg-SCST failed to describe any correct information. In contrast, PaG-MEG-SCST trained with MEG is able to describe “a bunch of people” sitting on “a wooden boat”, with “umbrellas on the ground”. Or in image 7 where the generation model trained without MEG detected a soccer ball as “frisbee”.

With the use of MEG during training, the generation model tends to try describing an image with various details. For example, image 4, it describes the “brown dog”, something “brown in the water behind the dog”, and the dog’s collar. Although, some property might not be accurate (e.g., “black collar” - image 4), at least the model try to include that level of details when generating the description. In some cases, the wrong information generated might be due to the word

association issue of language model. For example, “the water is blue” (image 3), or “the sky is blue” (image 2) are incorrect for these particular images, but usually they are correct. The model can be biased towards a specific property for a particular object due to the training data we used for both training the captioning model and pretrained models. This issue is very interesting and challenging since a model must have the capability of understanding and incorporating visual and textual information, e.g., really “know” or “understand” the meaning of colors for both visual and textual modalities.

## VI. DISCUSSION AND CONCLUSION

In this paper, we present a novel multimodal encoding generator, MEG, which generates encoding conditioned on visual (input image), textual (the sentence), and sequential information (the corresponding paragraph). We showed that MEG is useful for the task of image paragraph captioning when being used as a regularizer during training the paragraph generation model, PaG. In particular, we minimize the distance between the corresponding encoding vectors generated by PaG and MEG, and initialize PaG’s decoder with the pretrained MEG’s decoder. This regularization technique and the training strategy are generic and improve image paragraph generation performance in all the evaluation metrics. In addition, the training time for the paragraph captioning model to obtain the best performance is also shortened. MEG effectively captures the textual and visual information in a joint space, thus provides useful information for training the generation model. The qualitative experiments show that multimodal encoding generated by MEG also contains some level of semantic information. Our multimodal encoding is a visual-linguistic representation and we have used it for other tasks such as text-query based image retrieval and paragraph based instance retrieval tasks as well. Our model was able to obtain satisfactory results for paragraph-based image retrieval and paragraph-based instance retrieval tasks. We also observed that the length of the generated paragraph plays a huge role in the performance. Specifically, the deviation from ground truth length impacts the CIDEr score very badly while the METEOR score is stable.

In the future work, we aim to explore multimodal self-supervised and semi-supervised learning using MEG and PaG models. Furthermore, we aim to extend this work for visual inspection based diagnosis report generation tasks.

## ACKNOWLEDGEMENT

This research/project is supported in part by the National Research Foundation Singapore and DSO National Laboratories under the AI Singapore Programme (AISG Award No: AISG2-RP-2020-016) and by A\*STAR under its Knowledge Extraction, Modelling, and Explainable Reasoning for General Expertise (K-EMERGE) programme (Grant number A19E2b0098). This research is also supported by funding allocation to B.F. by the Agency for Science, Technology and Research (A\*STAR) under its SERC Central Research Fund (CRF), as well as its Centre for Frontier AI Research (CFAR).

No.	Query	Text-based PaG-NoReg	Text-based PaG-MEG	Encoding-based
1	A blue double decker bus is traveling down a paved road on a sunny day. The bus is decorated with balloons, banners and other decorations. People are standing on the sides of the road watching the bus. Tall buildings can be seen behind and on the side of the bus. Part of a leafless tree can be seen behind the bus. A group of people are walking behind the bus on the road in the background of the photo.			
2	A man is holding the string of a flying kite. The man is wearing a gray hat and a gray and light gray jacket. The kite is light green with two eyes, a nose, and mouth and two red spots drawn on it. The kite is a triangle shape and has three green strips hanging from the bottom of the kite. Four other kites can be seen in the sky. The sky is full of white and gray clouds, with a patch of blue. Several trees are seen in the lower portion of the image with a light pole with two lights on it close by.			
3	A snowboarder is high in the air after a jump off of a steep ramp at an event. The boarder has on a colorful coat, and black pants with a black vest over the coat with the number 48 on the vest. There is a constructed jump ramp on the right with a black banner with various brands written on it. The sky above is bright blue, and clear.			
4	An airplane is flying in the sky. The sky is clear and a dark blue color. The airplane is white and had a dark blue underside. There are four small black tires on the bottom of the airplane. It is a large airplane with many windows on the side of it. There are two dark blue circle jets on each side of the wings of the airplane.			

TABLE VIII: Image instance retrieval examples (qualitative results)

No.	Image	Ground-truth	PaG-NoReg-SCST	PaG-MEG-SCST
1		A bed is sitting inside of a room with white painted walls. The bed is sitting on a brown hardwood floor. White covers with colored flowers are on top of the bed. A white table is sitting near the bed. Different objects are sitting on top of the table. A window is sitting above the table. Green bushes can be seen outside of the window.	a <b>bedroom</b> is <b>sitting</b> in a room. there is a white in front of the bed. <b>there is</b> a white in front of the bed. <b>there is</b> a window on the wall next to the bed. there is a window on the wall next to the bed. there is a window on the wall next to the bed. there is a window on the wall next to the bed.	a bed is in a room. there is a window on the wall next to the bed. there is a white pillows on top of the bed. the bed is a red and white. there is a wooden on top of the bed. there is a small window on the side of the bed. the floor is a blue and in front of the bed.
2		Two airplanes are flying in the sky very close to each other. They are flying towards each other and look close enough to collide. The airplanes are both white with red stripe on the bottom half of the fuselage. The planes have small red engines on its wings. The sky is white.	a bunch of airplanes are parked on the runway. there is a white and on the side of the plane. there is a white and on the side of the plane. there is a white and on the side of the planes. the sky is blue and on the side of the planes.	two airplanes are flying in the sky. the planes are blue. the people on the plane is white. the sky is blue and white. the tail of the planes are white. the plane on the left of the plane is a black and white. to the sky is blue and there are in the background.
3		There is a male wearing a grey and black wet suit standing on a white surfboard. The surfboard is tilted backward and the person has his hands raised to balance and catch himself. There are three other people that are in the water in the background. The water with gray colored and foamy with small waves crashing toward shore.	a man is standing on a surfboard. the man is wearing a black wet suit in front of the water. the man is wearing a black wet suit in front of the water. the man is wearing a black wet suit in front of the water. the man is wearing a black and white in the water.	a man is surfing on a surfboard. there is a large wave in the water in the water. there is a man in a white surfboard. the man is wearing a black wet suit. the man is on the top of the wave the water is blue and there are waves in the ocean.
4		A large brown dog is standing in the water. There are small waves in the water around the dog. The dog is wearing a red collar around its neck.	a dog is standing on a beach. the dog is wearing a black and white in front of the water. the dog is wearing a black and white in front of the water. the dog has a black and white in front of the water. the dog has a black and white in front of the water.	a brown dog is standing in the water. there is a brown in the water behind the dog. the dog is wearing a black collar. the dog has a brown and white. the dog has a red on top of the dog. there is a large green grass on the side of the dog.
5		A canal with several boats in it. There are colorful umbrellas above the boats. There is a wooden ceiling above the boats.	a bunch of people are sitting on a beach. there are people standing on the beach. there are people standing on the beach. there are people standing on the beach. there is a small on the side of the kites. there is a small on the side of the kites. there is a small on the side of the kites.	a bunch of people are sitting on a wooden boat. there are umbrellas on the ground behind the boats. there is a large boat in front of the boats. the boats are multicolored and white. the umbrellas are sitting on top of the boats. there is a large green on the side of the water.
6		A woman in a white shirt and red skirt is playing tennis. She is swinging a tennis racket at a ball. There is a black fence behind the woman.	a woman is standing on a tennis court. she is wearing a white shirt and white shorts. she is wearing a white shirt and white shorts. the woman is wearing a white shirt and white shorts. the woman is wearing a white shirt and white shorts. the woman is wearing a white shirt and white shorts. the woman is wearing a white shirt and white shorts.	a woman is playing tennis on a tennis court. she is wearing a white shirt and white shorts. she is holding a tennis racket in her hands. the girl is wearing a black and white. the girl is holding a red on the side of the tennis court. there is a little trees on the wall behind the fence.
7		A man is jumping in the air to catch a soccer ball. There is another man standing in front of him. There is a silver fence behind them.	a group of people are playing frisbee on a frisbee. there is a man in a white shirt and white shorts. there is a man in a white shirt and white shorts. the man is wearing a white shirt and white shorts. the catcher is wearing a white and white in front of the bat. the man is wearing a white shirt and white shorts.	a man is standing on a soccer ball. there is a man in a white shirt and white shorts. he is holding a white ball in his hand. the man is wearing a black and white. the baseball player is holding a red on the side of the field there is a large green grass on the side of the field.

TABLE IX: Paragraphs generated by the paragraph captioning models with and without ViRAE (both fine-tuned using reinforcement learning). Ground-truth paragraphs are displayed for references. With additional information from ViRAE during training, V2P<sub>ViRAE</sub> is able to describe the images in more details and reduce generating repeated sentences. Words in colors show if the generated content is **correct**, **wrong**, **unsure**, or **repeated sentence** (best viewed in colors).

## REFERENCES

- [1] Q. You, H. Jin, Z. Wang, C. Fang, and J. Luo, "Image captioning with semantic attention," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.
- [2] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *International conference on machine learning*. PMLR, 2015, pp. 2048–2057.
- [3] J. Krause, J. Johnson, R. Krishna, and L. Fei-Fei, "A hierarchical approach for generating descriptive image paragraphs," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 317–325.
- [4] S. Aditya, Y. Yang, and C. Baral, "Explicit reasoning over end-to-end neural architectures for visual question answering," in *Thirty-second AAAI conference on artificial intelligence*, 2018.
- [5] N. Vijayaraju, "Image retrieval using image captioning," Master's thesis, San José State University, 2019.
- [6] F. Ahmed, M. S. Mahmud, R. Al-Fahad, S. Alam, and M. Yeasin, "Image captioning for ambient awareness on a sidewalk," in *2018 1st International Conference on Data Intelligence and Security (ICDIS)*. IEEE, 2018, pp. 85–91.
- [7] O. Sidorov, R. Hu, M. Rohrbach, and A. Singh, "Textcaps: a dataset for image captioning with reading comprehension," in *European Conference on Computer Vision*. Springer, 2020, pp. 742–758.
- [8] X. Liang, Z. Hu, H. Zhang, C. Gan, and E. P. Xing, "Recurrent topic-transition gan for visual paragraph generation," in *Proceedings of the IEEE international conference on computer vision*, 2017.
- [9] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [10] S. Wu, Z.-J. Zha, Z. Wang, H. Li, and F. Wu, "Densely supervised hierarchical policy-value network for image paragraph generation," in *IJCAI*, 2019, pp. 975–981.
- [11] Y. Mao, C. Zhou, X. Wang, and R. Li, "Show and tell more: Topic-oriented multi-sentence image captioning," in *IJCAI*, 2018.
- [12] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *The Journal of machine Learning research*, vol. 3, pp. 993–1022, 2003.
- [13] Y. Luo, Z. Huang, Z. Zhang, Z. Wang, J. Li, and Y. Yang, "Curiosity-driven reinforcement learning for diverse visual paragraph generation," in *Proceedings of the 27th ACM International Conference on Multimedia*, 2019, pp. 2341–2350.
- [14] Z.-J. Zha, D. Liu, H. Zhang, Y. Zhang, and F. Wu, "Context-aware visual policy network for fine-grained image captioning," *IEEE transactions on pattern analysis and machine intelligence*, 2019.
- [15] S. J. Rennie, E. Marcheret, Y. Mroueh, J. Ross, and V. Goel, "Self-critical sequence training for image captioning," in *CVPR*, 2017, pp. 7008–7024.
- [16] L. Melas-Kyriazi, A. M. Rush, and G. Han, "Training for diversity in image paragraph captioning," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2018, pp. 757–761.
- [17] S. Yan, Y. Hua, and N. Robertson, "Paracnn: Visual paragraph generation via adversarial twin contextual cnns," *arXiv:2004.10258*, 2020.
- [18] X. Yang, H. Zhang, and J. Cai, "Auto-encoding and distilling scene graphs for image captioning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.
- [19] Z. Wang, Y. Luo, Y. Li, Z. Huang, and H. Yin, "Look deeper see richer: Depth-aware image paragraph captioning," in *Proceedings of the 26th ACM international conference on Multimedia*, 2018, pp. 672–680.
- [20] W. Che, X. Fan, R. Xiong, and D. Zhao, "Paragraph generation network with visual relationship detection," in *Proceedings of the 26th ACM international conference on Multimedia*, 2018, pp. 1435–1443.
- [21] X. Yang, C. Gao, H. Zhang, and J. Cai, "Hierarchical scene graph encoder-decoder for image paragraph captioning," in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, pp. 4181–4189.
- [22] A. Gupta, Z. Shen, and T. Huang, "Text embedding bank for detailed image paragraph captioning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 18, 2021, pp. 15 791–15 792.
- [23] Y. Song, S. Chen, and Q. Jin, "Towards diverse paragraph captioning for untrimmed videos," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2021, pp. 11 245–11 254.
- [24] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015.
- [25] C. Yang, L. Xie, C. Su, and A. L. Yuille, "Snapshot distillation: Teacher-student optimization in one generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2859–2868.
- [26] J. Gou, B. Yu, S. J. Maybank, and D. Tao, "Knowledge distillation: A survey," *International Journal of Computer Vision*, vol. 129, no. 6, pp. 1789–1819, 2021.
- [27] T. Yao, Y. Pan, Y. Li, Z. Qiu, and T. Mei, "Boosting image captioning with attributes," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 4894–4902.
- [28] Z. Gan, C. Gan, X. He, Y. Pu, K. Tran, J. Gao, L. Carin, and L. Deng, "Semantic compositional networks for visual captioning," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 5630–5639.
- [29] C. Gan, Z. Gan, X. He, J. Gao, and L. Deng, "Stylenet: Generating attractive visual captions with styles," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 3137–3146.
- [30] X. Duan, W. Huang, C. Gan, J. Wang, W. Zhu, and J. Huang, "Weakly supervised dense event captioning in videos," *Advances in Neural Information Processing Systems*, vol. 31, 2018.
- [31] X. Long, C. Gan, and G. De Melo, "Video captioning with multi-faceted attention," *Transactions of the Association for Computational Linguistics*, vol. 6, pp. 173–184, 2018.
- [32] J. Mao, C. Gan, P. Kohli, J. B. Tenenbaum, and J. Wu, "The neuro-symbolic concept learner: Interpreting scenes, words, and sentences from natural supervision," *arXiv preprint arXiv:1904.12584*, 2019.
- [33] C. Gan, Y. Li, H. Li, C. Sun, and B. Gong, "Vqs: Linking segmentations to questions and answers for supervised attention in vqa and question-focused semantic segmentation," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 1811–1820.
- [34] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *Advances in neural information processing systems*, vol. 28, pp. 91–99, 2015.
- [35] T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Lisbon, Portugal: Association for Computational Linguistics, Sep. 2015, pp. 1412–1421. [Online]. Available: <https://www.aclweb.org/anthology/D15-1166>
- [36] R. Lin, S. Liu, M. Yang, M. Li, M. Zhou, and S. Li, "Hierarchical recurrent neural network for document modeling," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 2015, pp. 899–907.
- [37] M. Chatterjee and A. G. Schwing, "Diverse and coherent paragraph generation from images," in *ECCV*, 2018, pp. 729–744.
- [38] J. Wang, Y. Pan, T. Yao, J. Tang, and T. Mei, "Convolutional auto-encoding of sentence topics for image paragraph generation," in *International Joint Conferences on Artificial Intelligence*, 2019.
- [39] P. Anderson, X. He, C. Buehler, D. Teney, M. Johnson, S. Gould, and L. Zhang, "Bottom-up and top-down attention for image captioning and visual question answering," in *CVPR*, 2018, pp. 6077–6086.
- [40] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Empirical Methods in Natural Language Processing (EMNLP)*, 2014. [Online]. Available: <http://www.aclweb.org/anthology/D14-1162>
- [41] W. Zaremba, I. Sutskever, and O. Vinyals, "Recurrent neural network regularization," *arXiv preprint arXiv:1409.2329*, 2014.
- [42] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint:1412.6980*, 2014.
- [43] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, 2002, pp. 311–318.
- [44] M. Denkowski and A. Lavie, "Meteor universal: Language specific translation evaluation for any target language," in *Proceedings of the ninth workshop on statistical machine translation*, 2014, pp. 376–380.
- [45] R. Vedantam, C. Lawrence Zitnick, and D. Parikh, "Cider: Consensus-based image description evaluation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 4566–4575.
- [46] X. Chen, H. Fang, T.-Y. Lin, R. Vedantam, S. Gupta, P. Dollár, and C. L. Zitnick, "Microsoft coco captions: Data collection and evaluation server," *arXiv preprint arXiv:1504.00325*, 2015.
- [47] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European conference on computer vision*. Springer, 2014, pp. 740–755.
- [48] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of machine learning research*, vol. 9, no. 11, 2008.