



Geometry Attention Transformer with position-aware LSTMs for image captioning[☆]

Chi Wang¹, Yulin Shen¹, Luping Ji^{*}

School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, PR China

ARTICLE INFO

Keywords:

Image captioning
Transformer framework
Gate-controlled geometry attention
Position-aware LSTM

ABSTRACT

In recent years, Transformer structures have been widely applied in image captioning with impressive performance. However, previous works often neglect the geometry and position relations of different visual objects. These relations are often thought of as crucial information for good captioning results. Aiming to further promote the image captioning by Transformers, this paper proposes an improved *Geometry Attention Transformer* (GAT) framework. In order to obtain geometric representation ability, two novel geometry-aware architectures are designed respectively for the encoder and decoder in our GAT by i) a geometry gate-controlled self-attention refiner, and ii) a group of position-LSTMs. The first one explicitly incorporates relative spatial information into the image representations in encoding steps, and the second one precisely informs the decoder of relative word positions for generating caption texts. The image representations and spatial information are extracted by a pretrained Faster-RCNN network. Our ablation study has proved that these two designed optimization modules could efficiently improve the performance of image captioning. The experiment comparisons on the datasets MS COCO and Flickr30K, also show that our GAT could often outperform current state-of-the-art image captioning models.

1. Introduction

Image captioning is a challenging problem in computer vision (Farhadi et al., 2010). It aims to automatically describe an image using meaningful text, translating an image into natural language. It often requires to not only recognize what visual objects an image contains, but also accurately capture what those objects are doing, and even tell us what the interrelations of different objects are. Image captioning could build a powerful bridge between visual images and human languages. With captions, people could better understand an image (Krishna et al., 2017). Image captioning has been thought of as one quite useful technology in image analysis and understanding. In recent years, it has been attracting more and more research attention, so that diverse models have appeared for image captioning (Anderson et al., 2018; Cornia, Stefanini, Baraldi, & Cucchiara, 2020; Rennie, Marcheret, Mroueh, Ross, & Goel, 2017; Zhang, Li, Wang, Zhao and Wang, 2021).

For image captioning, its early models could be roughly classified into two primary categories (Bai & An, 2018). One category depends on the image retrieval by analyzing image correlation and then retrieving candidate texts from existing caption pools (Gupta, Verma, & Jawahar,

2012; Ordonez, Kulkarni, & Berg, 2011). The other is summarized as a template-based category. This kind of methods builds captions typically through the syntactics and semantics analysis on images, with visual concept detecting, sentence template matching and optimizing (Kulkarni et al., 2013; Socher & Fei-Fei, 2010; Ushiku, Yamaguchi, Mukuta, & Harada, 2015). The most obvious characteristic of these two early categories is that hard-coded rules and manually-designed features are in common use, so that the reliability and accuracy of captioning could often fluctuate heavily (Bai & An, 2018).

In recent years, facilitated by booming artificial intelligence, neural networks, the third category of models specially for image captioning, have been becoming one of the most exciting and powerful tools. For example, we could easily see the obvious captioning performance improvements on early algorithms, obtained by the deep neural models in Karpathy, Joulin, and Li (2014), Ma, Lu, Shang, and Li (2015) and Yan and Mikolajczyk (2015). Besides, more neural network-based models, such as the multimodal learning (Karpathy & Fei-Fei, 2015, 2017), the encoder-decoder framework (Vinyals, Toshev, Bengio, & Erhan, 2015), the attention mechanism (Huang, Wang, Chen, & Wei, 2019; You, Jin, Wang, Fang, & Luo, 2016), the compositional architectures (Oruganti,

[☆] Source code is available at <https://github.com/UESTC-nnLab/GAT>.

^{*} Corresponding author.

E-mail addresses: chi.w@std.uestc.edu.cn (C. Wang), yulinshen@std.uestc.edu.cn (Y. Shen), jiluping@uestc.edu.cn (L. Ji).

¹ These authors contributed equally to this work.

Sah, Pillai, & Ptucha, 2016), the describing method of novel objects (Mao et al., 2015) and the deep bifurcation network (Nabati & Behrad, 2021), have also been proposed one after another.

Through the survey of a large number of publications, it is easy to figure out that the Transformer with an encoder–decoder structure is becoming one of the most mainstream models for image captioning due to its outstanding performance (Chen, Wang, Yang and Li, 2021; Guo et al., 2020; Herdade, Kappeler, Boakye, & Soares, 2019; Zhou et al., 2020). Generally in a typical Transformer model, a set of intermediate vectors are extracted from a given image by an encoder with CNN-based networks, and then the target caption of this image is generated word by word through RNN-based decoder networks (Anderson et al., 2018; Karpathy & Fei-Fei, 2015; Lu, Xiong, Parikh, & Socher, 2017; Vinyals et al., 2015).

In addition, classic attention mechanism has also been widely utilized in image captioning (Huang et al., 2019; You et al., 2016). It could often effectively guide decoders to focus on some specific information, such as context correlation, for generating image captions. The captioning models with the self-attention mechanism (Vaswani et al., 2017) are often believed to be more effective than their early versions. In recent years, this kind of self-attention models have also quickly emerged in image captioning (Herdade et al., 2019; Li, Zhu, Liu, & Yang, 2019).

From the task view of image captioning, the geometry and position relations of image objects are necessary to accurately describe an image. For example, “a boy standing on a skateboard” and “a boy raises a skateboard in his hands” represent two different semantics, though they both contain the prominent visual objects “boy” and “skateboard” in an image. Therefore, the relative geometry information plays a key role in describing images. In most of existing Transformer models with attention modules, we could find that the inherent geometry and position relations between image components have not yet be paid enough attention to and fully utilized (Lu et al., 2017). Moreover, although some methods, such as the GSA proposed in Guo et al. (2020), carefully concern geometry relations in encoder, more precise position decoding has not been well solved yet.

Geometry and position relations are so important that accurate image captioning models have to emphasize them. In order to take full advantage of geometry and position clues, we first design an improved encoder model with Geometry Self-attention Refiner (GSR) in our Transformer architecture. This model could explicitly incorporate geometry information into Vanilla self-attention module, implementing the transforming from original appearance queries to geometrical ones. And then, the refined attention weights are calculated to get a mean of weighted appearance values. The geometrical queries and keys in our model are linearly derived from the bounding box coordinates of each visual object. In this way, every object could always obtain its appearance features and the geometry correlations with others. These points are crucial for a decoder to generate the right word sequences with position and semantics.

Furthermore, in the decoder of a Transformer architecture, it is unable to handle the sequence order of input tokens at structure levels, due to its parallel mechanism. It treats each token equally to facilitate parallelism while ignoring their sequence order in output text. Some methods inject the word position information of expected caption texts by adding a *sine* or *cosine* operation on the top of word embedding layers, such as the methods in Herdade et al. (2019) and Vaswani et al. (2017). This kind of position injection mechanism is proved effective, in spite of not always conforming to semantics. To overcome this weakness of position decoding, we design a group of position-LSTMs to model the word order of caption texts. It parses a caption sentence word by word in sequence, ensuring the right order of words. In addition, the hidden layers of position-LSTMs could also remember the information that decoder has generated. This kind of design is helpful to self-attention modules to concentrate on particular position

parts. In the meanwhile, the information, stored in hidden states of LSTMs, also contains the geometry information of visual objects.

Finally, by the integration and collaborative optimization of the GSR and the position-LSTM, our captioning model has achieved better performance than the state-of-the-art ones on the datasets MS COCO and Flickr30k. In summary, the main contributions of this paper could be listed as follows:

(i) An improved image captioning model, GAT is proposed on the base of the traditional Transformer framework. It could reliably capture the geometry relations of visual objects. Due to this merit, our model could often obtain such a sense of ‘where the target objects are’ and ‘where the captioning model is currently looking at’.

(ii) We design the encoders cooperated with a gate-controlled self-attention refiner. It could efficiently encapsulate the relative geometry information of objects, so as to further refine visual object representations.

(iii) We combine the decoders with a group of position-LSTMs. The position-LSTM could deliver the word order of generated caption text with relative position relations to the decoder. Therefore, it could decide which word to generate next by a trained position reasoning logic.

(iv) A group of ablation experiments, and two groups of offline/online comparison experiments are designed on datasets MS COCO and Flickr30K. These experiments prove the superiority of our GAT model in image captioning.

2. Related work

The encoder–decoder architecture of Transformer is widely used in many sequence-to-sequence problems, such as machine translation. Given a group of tokens as input, the encoders extract mutual information among these tokens and then generate new representations which contain the context of these tokens. A group of decoders will decode the intermediate vectors into a new sequence. The function of encoder–decoder structure is to map an input sequence into its target domain. To handle the sequence problem, the neural layers of encoder and decoder are often designed by RNN, LSTM or self-attention layers in previous works. Fig. 1 illustrates a typical encoder–decoder Transformer architecture with self-attention layers.

In recent years, the Transformer, a kind of encoder–decoder architecture with attention modules has been attracting wide research enthusiasm in image captioning. As a result, more and more approaches are developed for applications, such as the model by the semantic alignment (Lu, Guo, Dai, & Wang, 2022), the method with residual connections (Gao et al., 2019), the one with meshed-memory Transformer (Cornia et al., 2020) and the multi-stage aggregated Transformer (Zhang et al., 2021).

The classic architecture of Transformer for image captioning (Chen, Wang et al., 2021; Guo et al., 2020; Herdade et al., 2019; Zhou et al., 2020), could be clearly seen in Fig. 1. It consists of an encoder and a decoder, one for extracting the mutual features of visual objects and the other for generating caption texts. Fig. 1(a) shows the encoder–decoder structure of the Transformer (Yan et al., 2022), whose layers are designed by the self-attention modules with residual connections (Herdade et al., 2019; Li et al., 2019). A classic self-attention module is shown in Fig. 1(b).

In the publication (Guo et al., 2020), shown in Fig. 1, for the input to the encoder, besides traditional object appearance features, it also contains some novel geometry cues, such as object center, height and width. As for the decoder, a position encoding operation has also been employed to encode the accurate word sequence of caption text. A simple sine function is utilized to fulfill this task. The experiments in Guo et al. (2020) proved that this kind of improvement by geometry and position features is efficient. To our knowledge, this kind of encoder–decoder architecture with attention components is believed to be one

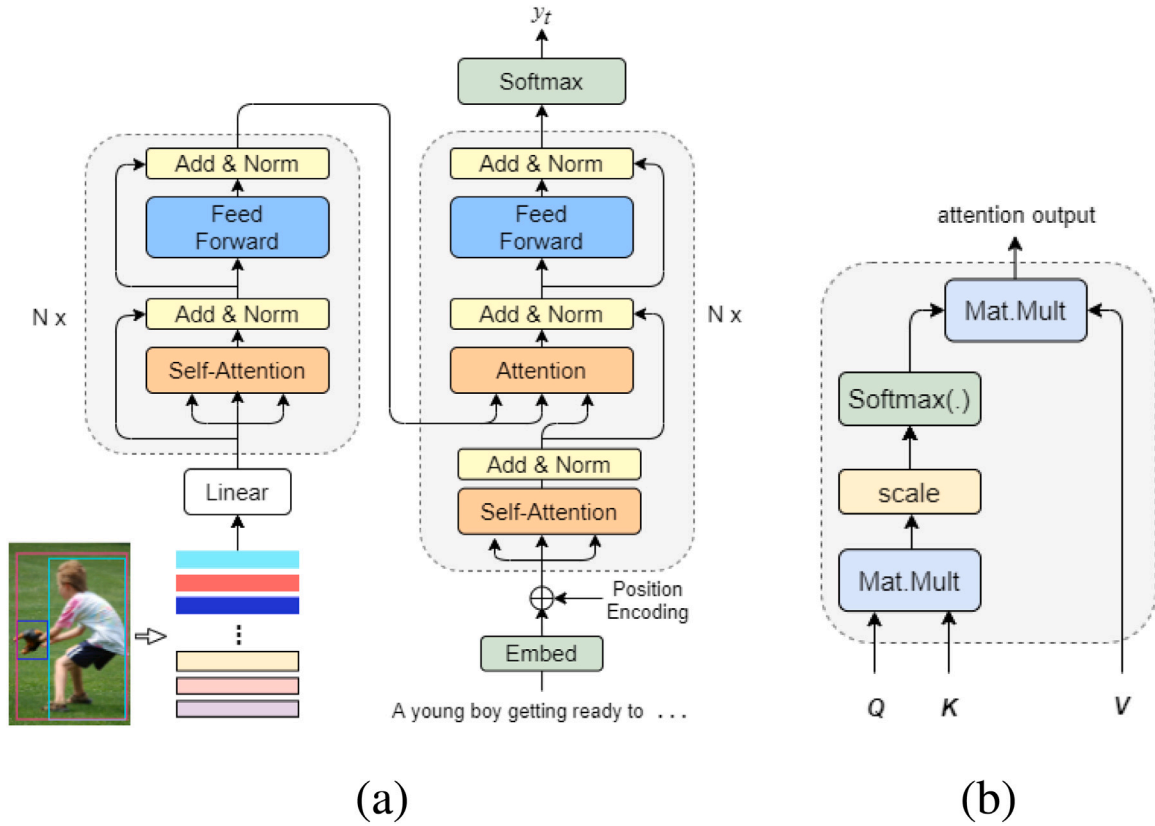


Fig. 1. (a) A typical Transformer framework for image captioning (Guo et al., 2020). N means the number of layers, and y_t is the embedding of the t th word in a generated sequence. (b) A classic attention module. Q , K and V represent the Query matrix, the Key matrix and the Value matrix, respectively. Query could get its corresponding Value by the weighted average of the similarities between two matrices Query and Key. In a self-attention module, it usually keeps $Q = K = V$.

of the most powerful baselines for image captioning (Chen, Wang et al., 2021; Herdade et al., 2019; Xian, Li, Zhang, & Ma, 2022).

Moreover, in terms of the attention modules in image captioning, there are all kinds of variants having been explored. For example, we could see the multi-head attention in Vaswani et al. (2017), the fully attentive paradigms (Li et al., 2019; Zhu, Li, Liu, Peng, & Niu, 2018), the meshed-connection attention (Cornia et al., 2020), the dual attention (Yu, Zhang, & Wu, 2022), the RSTNet (Zhang et al., 2021) and the task-adaptive attention (Yan et al., 2022).

In the future, more ingenious network design, more sophisticated object features, more functional geometry relations, and more efficient position encoding strategies, are believed to be mainstream research expectations for the further performance promotion of image captioning methodologies (Chen, Jiang and Zhao, 2021; Liu, Ren, & Yuan, 2021; Shen & Wang, 2022; Xian et al., 2022).

3. Proposed method

Our method generates grounded captions by attending to specific image regions at each step. In terms of structure, it still retains a classic encoder-decoder architecture. In its encoder, a geometry self-attention refiner is designed to optimize image representations. And in its decoder, a module for accurately decoding word sequences by LSTM is adopted. More details about it can be seen in Fig. 2.

Given an input image I , in Fig. 2, we use $(X_A \in \mathbb{R}^{N \times d})$ to represent a set of image appearance regions, where N is the index of image region, and d is the dimensionality of region data vector. Moreover, $y = \{y_1, \dots, y_T\}$ indicates a group of caption word vectors corresponding to the image I .

3.1. Gate-controlled geometry self-attention refiner

Besides appearance information X_A , we incorporate geometry information into Vanilla self-attention to refine the representation of image objects, for the reason that it is beneficial to comprehend the intrinsic relations among different visual objects. Therefore, we propose the GSR to refine the visual information by taking the geometry information of visual objects into consideration.

Given the geometry features of the objects $X_g \in \mathbb{R}^{N \times 5}$, each row of X_g is a 5-dimensional vector, as follows:

$$(x_{min}, y_{min}, x_{max}, y_{max}, S) \quad (1)$$

where (x_{min}, y_{min}) , (x_{max}, y_{max}) and S represent the top-left coordinates, the bottom-right coordinates of bounding box, and the relative size to whole image, respectively. Moreover, all of them are normalized to $(0, 1)$.

Firstly, we obtain $X_G \in \mathbb{R}^{N \times d_m}$ by embedding X_g into a higher dimension form with an embedding layer followed by a ReLU non-linearity operation. Then we combine appearance information with geometry-related information by modifying the queries and keys as follows:

$$Q' = [X_A W_{Q_A}; X_G W_{Q_G}] \quad (2)$$

$$K' = [X_A W_{K_A}; X_G W_{K_G}] \quad (3)$$

where W_{Q_A} , W_{K_A} are two learned appearance matrices and W_{Q_G} , W_{K_G} are two learned geometry matrices, respectively. And all of them share the dimensionality of $\mathbb{R}^{d_m \times d_m}$. Here, $[\cdot]$ indicates an operator of concatenation. Q' and K' combine the appearance information of objects with their geometry relations. This could be seen as a complementary means to acquire the fine-grained knowledge of image objects. In details, the

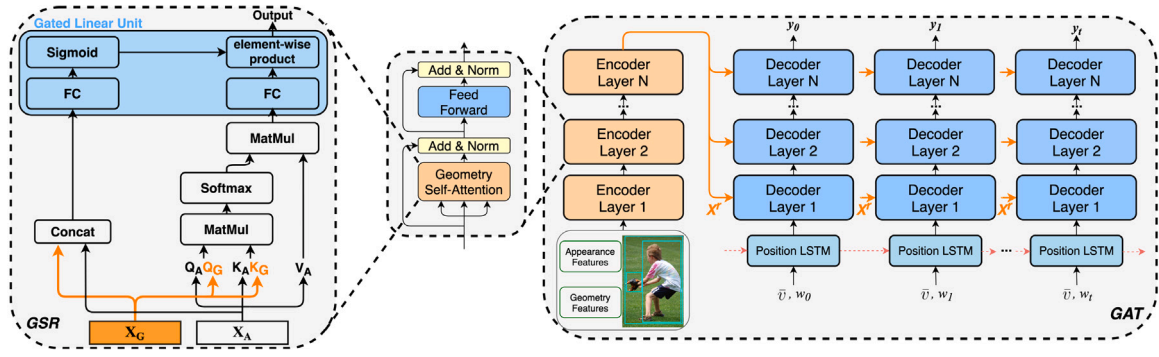


Fig. 2. The detailed architecture of our GAT. The GSR module in encoder generates the weighted mean of appearance features X_A and geometry features X_G . It is further refined by a gate-controlled unit. Moreover, a group of position-LSTMs precisely deliver word order information into the decoders.

attention results of our model can be calculated by:

$$\Omega' = \frac{Q'K'^T}{\sqrt{2 \times d_k}} \quad (4)$$

Therefore, the output of the GSR in our model can be calculated by:

$$\text{Attention}_g(X) = \text{softmax}(\Omega')V_A \quad (5)$$

Similar to the original Transformer (Vaswani et al., 2017), our geometry refiner is also implemented in a multi-head self-attention framework. In this framework, the geometry attention will be repeated for h times (so, called h heads). In every time of repetition, it always adopts a group of different projection matrices ($W_{Q_A}^k, W_{K_A}^k$ and $W_{V_A}^k$) to extract appearance features, and takes ($W_{Q_G}^k$ and $W_{K_G}^k$) for to capture geometry features, where $k \in (1, 2, \dots, h)$. All the results from h self-attention heads will also be concatenated together by a linear projection, and then fed forward to their next layers.

Moreover, inspired by the work of Huang et al. (2019), we further design a parallelly Gate-controlled Linear Unit (GLU) to continually refine the output of geometry self-attention module (the left region of Fig. 2). This GLU still takes the current context ($\tilde{c} = [X_A; X_G]$) as its input to generate a control matrix G for modulating its final attention output by Hadamard product. This group of neural computation can be mathematically expressed by Eq. (6):

$$\begin{cases} \text{GateCtrl} = \text{sigmoid}(W_g \tilde{c} + b_g) \\ \text{Output} = \text{GateCtrl} \odot (W_i \tilde{a} + b_i) \end{cases} \quad (6)$$

where $W_g, W_i \in \mathbb{R}^{d_m \times d_m}$ are two groups of weights, and $b_g, b_i \in \mathbb{R}^{d_m}$ are two groups of neuron biases. Furthermore, \odot denotes a Hadamard product. In terms of structure, our self-attention component and gate-controlled unit could often be stacked into more multiple layers to further optimize object representations.

Subsequently, the self-attention output is embedded with its original input by the operation of *Adding & Normalization*, then transferred to its next layer, i.e., the *Feed-Forward* sub-layer network shown in the middle of Fig. 2. In details, this *Feed-Forward* sub-layer usually contains two groups of nested affine transformations with the activation function, $\text{ReLU}(x) = \max(0, x)$. Therefore, this group of processing steps could be uniformly formulated by

$$\text{FF}(X) = \max(0, XW_1 + b_1)W_2 + b_2 \quad (7)$$

where W_1 and W_2 are two groups of weight matrices to be learned, b_1 and b_2 are also two groups of biases. As in the Transformer architecture (Vaswani et al., 2017), each sub-layer is successively followed by residual connections and layer normalizing. Therefore, this group of computation could be mathematically expressed by

$$\begin{cases} Z = \text{LayerNorm}(X + \text{Attention}_g(X)) \\ X^r = \text{LayerNorm}(Z + \text{FF}(Z)) \end{cases} \quad (8)$$

3.2. Position-aware self-attention decoder

(i) Encoding positions via LSTM.

To address the word order problem of caption sequences, existing models usually inject some 'positional encoding' into the word embedding layer with a sine function. In our method, we further exploit an improved position representation strategy and then devise a distinctive LSTM-based position encoding mechanism.

The position-LSTM in our decoder of Transformer could model the order of image caption words in decoding process. For each time step t , we define the input of the position-LSTM as follows:

$$x_t = [w_t, \bar{v}] \quad (9)$$

where w_t is the word embedding derived by a one-hot vector, and $\bar{v} = \frac{1}{k} \sum_i v_i$ denotes the mean pooling of image features. Therefore, we could obtain:

$$h_t, c_t = \text{LSTM}(x_t, (h_{t-1}, c_{t-1})) \quad (10)$$

In view of the sequentiality output of LSTM, h_t could often be treated as an encoding of sequence order for caption words. Such an encoding could provide its subsequent decoder with precise position information in two aspects. One is what words have been generated by far, and the other is to where the decoder is directing the current relative position of objects. In addition, this position encoding will also be updated at each time step, and meanwhile it could also guide the decoder to adaptively focus on its correlated regions in geometry-aware mode.

(ii) Injecting position encodings into the decoder

Given a set of refined region features X^r and a group of current position encoding vectors h_t , our decoder could generate a multi-layer structure sequence \hat{y} . As shown in Eq. (7), each decoding layer of our model always contains a sub-layer of attention and a sub-layer of feed-forward output. Sometimes, such a layer may seem a little redundant. Therefore, in experiments we consider to remove one self-attention sub-layer so as to compress the network size of GAT, and then to further analyze the influence of removing attention sub-layer on captioning performance.

Furthermore, the group of decoders at bottom firstly achieve the current position encoding h_t to calculate the *Query*, and then use the region features X^r to calculate the *Keys* and *Values*. Then, the scaled attention results by dot-product could be obtained from a multi-head pattern. For the back layers of decoders, however, they always adopt the output of previous layers as the *Query*. As a result, at each time step t , through the conditioning operations on the output of top-layer decoder, the distribution probability of the t th word could be calculated by:

$$p(y_t | y_{1:t-1}) = \text{softmax}(W_p F_t + b_p) \quad (11)$$

where $W_p \in \mathbb{R}^{|\Sigma| \times m}$ and $b_p \in \mathbb{R}^{|\Sigma|}$ are also two groups of learnable parameters, and F_t is the final output of designed decoder in our model.

4. Experiments

4.1. Datasets and metrics

We evaluate our model on the MS COCO (Lin et al., 2014) and the Flickr30k (Young, Lai, Hodosh, & Hockenmaier, 2014). MS COCO is one of the largest datasets for image captioning, consisting of 123 287 images, with 5 ground-truth (GT) captions per image. For reliable validation and offline evaluation, we employ the widely-used “Karpathy” split (Karpathy & Fei-Fei, 2015), containing 113 287 images for training, 5000 for validation and 5000 for test. Flickr30k contains 31 783 images, also with 5 GT captions per image. For the fair comparisons with other methods, we still conform to the same data preparation in Karpathy and Fei-Fei (2015) and Yang, Tang, Zhang, and Cai (2019), i.e., truncating all the GT captions longer than 16 words and converting the remainders into lower cases. By doing so, we obtain an experimental vocabulary consisting of 9487 words from MS COCO and 7000 ones from Flickr30k, respectively.

Following public convention, there are five classic metrics to be adopted for evaluating the performance of image captioning. They consist of BLEU (Papineni, Roukos, Ward, & Zhu, 2002), METEOR (Banerjee & Lavie, 2005), ROUGE (Lin, 2004), CIDEr (Vedantam, Lawrence Zitnick, & Parikh, 2015) and SPICE (Anderson, Fernando, Johnson, & Gould, 2016). All these metrics can efficiently evaluate the captioning texts on their ground-truth captions. They focus on some different aspects of generated caption texts, such as adequacy and fluency. Please refer to the Appendix of this paper for more details.

4.2. Implementation details

We employ a pre-trained Faster R-CNN model with a backbone of ResNet-101 on dataset ImageNet to extract 36 features for each image. We use the same model weights offered in Anderson et al. (2018) for fair comparisons. The dimensionality of output feature vectors is 2048. And we project them into 512 dimensions to reduce memory consumption. The hidden size of LSTM is set to 1024. In addition, the dimensionality of the input to both the GSR and the self-attention module is set to 512. Moreover, the number of self-attention heads is 8. We set the number of layers to 3, in both the encoder and the decoder. In LSTM, the dropout rate is 0.5, and the dropout rate of all self-attention layers is set to 0.1. In the stage of cross-entropy training, we train our model using an initial learning rate of 5×10^{-4} with a decay rate of 0.8 for every 3 epochs. In CIDEr optimizing, we train our model in 30 epochs, by a learning rate of 2×10^{-5} with a decay factor of 0.8 also for every 3 epochs. All compared models are trained by the Adam optimizer with a batch size of 50. In tests, we always use a same beam size of five for all. Moreover, all models are trained on the training spilled of datasets, and evaluated on validation datasets to select the model with the best performance. Like these methods in Guo et al. (2020) and Huang et al. (2019), we also use a classic CIDEr metric to select our optimal model on validation datasets.

4.3. Ablation experiments

To quantify the performance of our new modules, we design a group of ablative experiments on MS COCO. We use the Vanilla Transformer (see Fig. 1) as our experiment ‘base’. It is similar to the self-attention network (Guo et al., 2020). Its encoders do not consider the geometry information, and decoders are only simply combined with common sine position encodings.

(i) Effect of Geometry Self-attention

We first apply the GSR to the ‘base’ model to evaluate its effect on encoders. Our GSR refines raw image representations by injecting explicit geometry relations with a multi-layer mode. From Table 1, we could see that it obtains an obvious improvement of CIDEr score from

Table 1

Ablation experiment comparisons. The results are reported after cross-entropy loss stage on the dataset “Karpathy” test split (Karpathy & Fei-Fei, 2015).

Model	BLEU-1	BLEU-4	METEOR	ROUGE	CIDEr	SPICE
Base	75.0	32.8	27.3	55.5	109.0	20.6
Base+GSR	76.9	35.6	28.1	57.0	115.1	21.4
Base+position-LSTM	76.5	34.5	28.0	56.8	114.9	21.3
Full: GAT	77.5	37.8	28.5	57.6	119.8	21.8

109.0 to 115.1 by applying our GSR. This comparison demonstrates that the base model without geometry relations could be confused by irrelevant regions and misled by wrong information. Admittedly, our GSR could furnish the base model with a sense of ‘where’. It means that our model could look purposefully, and thus it generates the caption with precise order and geometry-aware words.

(ii) Effect of Position LSTM

We further evaluate the efficiency of position-LSTM modules on image captioning. We replace the simple sine position encoding in Guo et al. (2020), with our LSTM-based ones. From the experimental comparisons in Table 1, we could see that our position-LSTM has raised the CIDEr score of ‘base’ model by 5.9. Compared with sine encoding, our position-LSTM could provide the decoder with an expressive encoding for ‘where’ to decode, and also guide the decoder to capture the correlated semantic information from image regions.

(iii) Geometry Queries and Keys

To verify the efficiency of concatenating geometrical queries and keys with appearance queries and keys, we investigate different strategies to combine them. Mainly, we directly compare two operations, including ‘add’ (adding) and ‘concat’ (concatenating). Table 2 shows the comparison results of these two operations. Compared with concatenating, we could see that, the performance promotion by concatenating appearance queries and keys with geometry queries and keys is better than by adding them, although they both could surpass the base model in terms of most metrics. For example, the BLEU-1 is only 75.0 by the base model, however it rises to 76.0 and 77.5 by ‘add’ and ‘concat’, respectively. Adding means that these two groups of different queries and keys share one set of common weights and use a same update rate. It is often not necessary in the case of image captioning.

(iv) Gate-controlled Linear Unit

In our architecture, the GLUs play an important role on refining the output of original self-attention layers, therefore we also have tested its effect on both encoders and decoders. Table 2 lists the comparison results. We could see that the GLUs could always achieve a good performance promotion on base model. However, we could also see that the GLU works best when it cooperates alone with encoders. Even, if our GLUs are assembled with encoders and decoders at the same time, on the contrary, they could even get worse. For example, in Table 2, the SPICE by the base model without GLUs is 21.2, and it gets to 21.8 when the GLUs are integrated only with encoders. However, the SPICE drops to 21.5 when the GLUs are combined with all encoders and decoders, exactly consistent with Huang et al. (2019). A probable explanation for this phenomena is that stacking too many GLUs on decoders could damage the gradient of network training, depressing the capacity of self-attention layers.

4.4. Numeric comparisons

We have also compared our GAT model with several state-of-the-art approaches of recent years. On dataset MS COCO, these comparison methods mainly include the SCST (Rennie et al., 2017) to directly optimize evaluate metrics, the Up-Down model (Anderson et al., 2018) with two-layer LSTM structure for extracting bottom-up features, and the ORT (Herdade et al., 2019) employing a Transformer-like model

Table 2

The comparison results of different component combination strategies, including geometry Queries and Keys, and the GLU module in the Transformer structure.

Modules	Strategy	BLEU-1	BLEU-4	METEOR	ROUGE	CIDEr	SPICE
With geometry Q & K	<i>add.</i>	76.0	35.1	27.2	56.0	116.4	20.7
	<i>concat.</i>	77.5	37.8	28.5	57.6	119.8	21.8
With GLU	Without GLU	76.7	35.2	27.6	56.5	114.9	21.2
	GLU(enc.)	77.7	37.8	28.4	57.4	119.4	21.8
	GLU(enc. and dec.)	77.4	37.8	28.2	57.1	118.9	21.5

Table 3

The offline comparisons with some state-of-the-art methods on the ‘Karpthy’ test splits of MS COCO. All data in columns, the higher the better performance. The visual features of compared models are extracted by a same structure with the Faster RCNN (Anderson et al., 2018). Our model has reduced the dimensionality of embedding from 2048 to 512, due to the limitation of main memory capability.

Model	Cross-Entropy loss						CIDEr-D optimization					
	BLEU-1	BLEU-4	METEOR	ROUGE	CIDEr	SPICE	BLEU-1	BLEU-4	METEOR	ROUGE	CIDEr	SPICE
Single model												
SCST (Rennie et al., 2017)	–	30.0	25.9	53.4	99.4	–	–	34.2	26.7	55.7	114.0	–
Up-Down (Anderson et al., 2018)	77.2	36.2	27.0	56.4	113.5	20.3	79.8	36.3	27.7	56.9	120.1	21.4
RFNet (Jiang et al., 2018)	76.4	35.8	27.4	56.8	112.5	20.5	79.1	36.5	27.7	57.3	121.9	21.2
GCN-LSTM (Yao et al., 2018)	77.3	36.8	27.9	57.0	116.3	20.9	80.5	38.2	28.5	58.3	127.6	22.0
SGAE (Yang et al., 2019)	–	–	–	–	–	–	80.8	38.4	28.4	58.6	127.8	22.1
ORT (Herdade et al., 2019)	–	–	–	–	–	–	80.5	38.6	28.7	58.4	128.3	22.6
AoANet (Huang et al., 2019)	77.4	37.2	28.4	57.5	119.8	21.3	80.2	38.9	29.0	58.8	129.8	22.4
GAT (ours)	77.5	37.8	28.5	57.6	119.8	21.8	80.8	39.7	29.1	59.0	130.5	22.9
Ensemble/Fusion												
SCST (Rennie et al., 2017) ^z	–	32.8	26.7	55.1	106.5	–	–	35.4	27.1	56.6	117.5	–
RFNet (Jiang et al., 2018) ^z	77.4	37.0	27.9	57.3	116.3	20.8	80.4	37.9	28.3	58.3	125.7	21.7
GCN-LSTM (Yao et al., 2018) ^z	77.4	37.1	28.1	57.2	117.1	21.1	80.9	38.3	28.6	58.5	128.7	22.1
SGAE (Yang et al., 2019) ^z	–	–	–	–	–	–	81.0	39.0	28.4	58.9	129.1	22.2
AoANet (Huang et al., 2019) ^z	78.7	38.1	28.5	58.2	122.7	21.7	81.6	40.2	29.3	59.4	132.0	22.8
GAT (Ours) ^z	79.0	38.8	28.7	58.9	123.7	22.1	81.6	40.7	29.4	59.6	133.4	23.2

with an object relation module. Besides, we also compare our model with the RF-Net (Jiang, Ma, Jiang, Liu, & Zhang, 2018) which employs features by multiple CNNs, the GCN-LSTM (Yao, Pan, Li, & Mei, 2018) exploiting pairwise relations by Graph Convolutional Network, the GAE (Yang et al., 2019) via auto-encoding scene graphs, the AoANet (Huang et al., 2019) refining self-attention results also by GLUs. On Flickr30k, the models compared with ours include the Soft-Attention & Hard-Attention (Xu et al., 2015), the Deep VS (Karpthy & Fei-Fei, 2015), the NIC (Vinyals et al., 2015), the m-RNN (Mao et al., 2014), the adaptive model (Lu et al., 2017), the SEM architecture (Cai & Liu, 2020) and the DA framework (Gao et al., 2019).

(i) Offline Evaluation

We also evaluate our model on the ‘Karpthy’ test split of MS COCO, as in Karpthy and Fei-Fei (2015). All models are firstly trained by cross-entropy loss, and then optimized on CIDEr scores. For fair comparisons, the visual features fed to all models are directly extracted by a Faster R-CNN of same structure. The top half of Table 3 (i.e., Single Model) shows the numeric comparisons of all models. It could be seen that our GAT outperforms all other methods on almost all metrics, in terms of cross-entropy loss and CIDEr Optimization. Compared with the ORT (Herdade et al., 2019), on CIDEr-D optimization, our GAT has obtained a BLEU-4 rise of 1.1%, METEOR rise of 0.4%, a ROUGE rise of 0.6%, CIDEr rise of 2.2%, and a SPICE rise of 0.3%. The rise values by single model seem quite obvious. Especially on BLEU-4, it achieves an increase from 38.6 to 39.7, and on CIDEr it rises from 128.3 to 130.5. Similarly, our GAT also outperforms the AoANet (Huang et al., 2019), in terms of almost all metrics except the CIDEr of Cross-Entropy.

Moreover, in the down half of Table 3 (i.e., the Ensemble/Fusion), it shows the comparisons of 6 different captioning models. Every model is tested for 4 times, respectively under 4 different initialization conditions, and then only the mean of 4 outputs is used for the comparisons of captioning models. We could also see that, our GAT could surpass all other models on almost all metrics of Cross-Entropy Loss. Even on the metrics of CIDEr-D optimization, our BLEU-1 could almost keep

the same as the AoANet (Huang et al., 2019), the best of all other models. On the rest metrics of CIDEr-D optimization, our GAT could always outperform all other models. For example, on the CIDEr of Cross-Entropy loss, our model achieves a result of 123.7, just 1.0 higher than the AoANet (Huang et al., 2019). And on the CIDEr of CIDEr-D optimization, our model rises to 133.4, however the AoANet (Huang et al., 2019) is only 132.0, approximately 1.4 less than ours.

In Table 4, we compare the performance of our GAT with state-of-the-art models on dataset Flickr30k. It could be seen that our model outperforms all the compared methods by a large margin. For example, compared with the one similarly employing an attention mechanism (Xu et al., 2015), our GAT obtains an improvement of 7.5 on BLEU-1, 10.9 on BLEU-4 and 4.9 on METEOR, respectively. Even compared with the DA model (Gao et al., 2019), the best of other eight models, our GAT still could often achieve higher score than the DA (Gao et al., 2019) on all metrics. For example, our GAT could get an increase of about 0.6, 1.4 and 1.4 on BLEU-1, BLEU-4 and CIDEr, respectively. This group of comparisons further demonstrate the superiority of our GAT method.

(ii) Online Evaluations

For fairer comparison, and knowing the true rank of our work in image captioning, we also submitted our GAT to an open test server specially for MS COCO. Our model is trained locally, updated to the website, and automatically evaluated by the cloud server. Note that online test data is not public, and nobody except official staff has the access authority to the test data. The last row in Table 5 lists the test results of our GAT model. In this test, an ensemble of 4 different outputs of our model trained by the ‘Karpthy’ split is used for the comparisons with others. From Table 5, we could see that our GAT obtains the best captioning performance on each c5 of almost all metrics, and it is a bit lower than others only on the c40 of three metrics (BLEU-1, METEOR and ROUGE). For example, it is 127.8 on CIDEr(c5), almost 0.9 higher than the best one of others, i.e., the AoANet (Huang et al., 2019). Even on BLEU-1(c40) and ROUGE(c40), our GAT is only a paltry 0.2 lower than the SGAE (Yang et al., 2019) and 0.1 lower than the

Table 4

The offline comparison results with state-of-the-art methods on dataset Flickr30k. The higher, the better for the data in all data columns.

	BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR	CIDEr
Deep VS (Karpathy & Fei-Fei, 2015)	57.3	36.9	24.0	15.7	15.3	24.7
Soft-Attention (Xu et al., 2015)	66.7	43.4	28.8	19.1	18.5	–
Hard-Attention (Xu et al., 2015)	66.9	43.9	29.6	19.9	18.5	–
Google NIC (Vinyals et al., 2015)	66.4	42.3	27.7	18.3	–	–
m-RNN (Mao et al., 2014)	60.0	41.0	28.0	19.0	–	–
Adaptive (Lu et al., 2017)	67.6	49.4	35.4	25.1	20.4	53.1
SEM (Cai & Liu, 2020)	73.1	55.1	40.1	29.0	22.0	66.8
DA (Gao et al., 2019)	73.8	55.1	40.3	29.4	23.0	66.6
GAT (ours)	74.4	56.7	41.8	30.8	23.4	68.0

Table 5

The leaderboard of various methods on the online test server specially for MS COCO. Our method, the GAT model, went into the top 10 list of leaderboard, when submitted online to <https://competitions.codalab.org/competitions/3221> on June 5th, 2020.

Model	BLEU-1		BLEU-2		BLEU-3		BLEU-4		METEOR		ROUGE		CIDEr	
	c5	c40	c5	c40	c5	c40	c5	c40	c5	c40	c5	c40	c5	c40
SCST (Rennie et al., 2017)	78.1	93.7	61.9	86.0	47.0	75.9	35.2	64.5	27.0	35.5	56.3	70.7	114.7	116.0
Up-Down (Anderson et al., 2018)	80.2	95.2	64.1	88.8	49.1	79.4	36.9	68.5	27.6	36.7	57.1	72.4	117.9	120.5
RFNet (Jiang et al., 2018)	80.4	95.0	64.9	89.3	50.1	80.1	38.0	69.2	28.2	37.2	58.2	73.1	122.9	125.1
GCN-LSTM (Yao et al., 2018)	–	–	65.5	89.3	50.8	80.3	38.7	69.7	28.5	37.6	58.5	73.4	125.3	126.5
SGAE (Yang et al., 2019)	81.0	95.3	65.6	89.5	50.7	80.4	38.5	69.7	28.2	37.2	58.6	73.6	123.8	126.5
AoANet (Huang et al., 2019)	81.0	95.0	65.8	89.6	51.4	81.3	39.4	71.2	29.1	38.5	58.9	74.5	126.9	129.6
NG-SAN (Guo et al., 2020)	80.8	95.0	65.4	89.3	50.8	80.6	38.8	70.2	29.0	38.4	58.7	74.0	126.3	128.6
GAT (Ours)	81.1	95.1	66.1	89.7	51.8	81.5	39.9	71.4	29.1	38.4	59.1	74.4	127.8	129.8

AoANet (Huang et al., 2019). Therefore, on the whole, our GAT model could be synthetically deemed to be more superior than the other ones.

4.5. Caption text comparisons

To visually illustrate the superiority of our model, in Fig. 3 we present ten typical images with the captions generated by our GAT, the AoANet (Huang et al., 2019) and the Vanilla Transformer (*i.e.*, the base), respectively. Moreover, the three ground-truth captions (*i.e.*, the GT1~GT3) of each image are also shown in Fig. 3.

It could be seen that the image captions with semantics and position relations, generated by our GAT, are often more precise than those by the AoANet (Huang et al., 2019). For example, It can be seen that our GAT could capture the visual objects “sheep” and “fence”, as well as the position relation (*i.e.*, “next to”) between “a herd of sheep” and the “wooden fence”. In contrast, the AoANet can also capture the “a group of sheep” and the “a fence”, however it does not precisely concern the material feature of “a fence”, *i.e.*, the “wooden”.

This advantage, that our GAT is able to accurately describe the geometry and position relations, mainly owes to the GSR module which explicitly incorporates the spatial correlations of image regions into object feature representations. Moreover, the position-LSTM could also keep reminding the decoders which visual object should be attended to, at every decoding step. Therefore, our GAT could often generate the highly-sufficient captions with spatial-aware semantics.

In addition, our experiments show that the GAT method with non-optimal parameters sometimes would over-emphasize the geometry relations of objects while making captioning sentences. In this case, the captions generated by our GAT could be less acceptable to human understanding. In our training and tests, some typical results are saved and could be seen in Fig. 4. For example, it generates the “sitting in the water” rather than the “sitting on the water” to describe the top left image in 4. Moreover, although our method tries to describe visual objects as many as possible and capture their geometry relations, it would also output some image captions with low fluency and semantic precision, such as the left one and the right one in the second row of Fig. 4.

Table 6

The comparisons of computation cost and parameter scale between our GAT and the baseline model. It shows that our model has fewer parameters and higher inference speed.

Captioning models	Parameter scale	Average time (per image)
Baseline model	70.6M	190 ms
Our GAT	37.0M	148 ms

4.6. Computation cost

Table 6 shows the computational burden of our GAT, compared to the base model mentioned in 4.3, *i.e.*, the Vanilla Transformer. The results are obtained on the platform with the GPU of Nvidia 2080 Super and the CPU of Intel 9900K.

Due to the high GPU memory consumption of LSTMs in model training, we decrease the number of encoder layers to 3, while the base model has 6 encoder layers. So the computational cost during the inference stage decreases, though we incorporate two new modules into our GAT model. Our method could usually obtain better performance with less inference time. For example, on two datasets the average captioning time is 148 ms by our GAT with only 37M parameters, however it is 190 ms by the base model with more than 70M parameters. The parameter quantity of our GAT model is approximately 52% of the base model, and the time cost of our GAT is only 77.8% of the base model. By comparisons, it could be believed that our model has the advantages of fewer parameters and faster caption generating speed.

5. Conclusions and future research

In this paper, we propose the Geometry Attention Transformer, an improvement and extension framework of the well-known Transformer for image captioning in recent years. Our model is able to explicitly refine image representations by incorporating the geometry features of visual objects into region encodings. Moreover, the position-LSTMs in decoder layers could also fulfill the precise encoding for the word order of caption texts. The ablation experiments on baseline model show that, the GSR for capturing geometry features and the position-LSTMs for injecting position encodings could often be effective. Each of them, if cooperating with a base model, could obviously promote image

	<p>GAT: A herd of <u>sheep</u> standing <u>next to</u> a wooden <u>fence</u>. Base: A group of sheep standing next to each other. AoANet: A group of sheep standing in front of a fence. GT1: Some sheep standing around by a wooden wall. GT2: A group of five sheep wait outside a barn. GT3: Five sheep are standing and sitting in their enclosure.</p>		<p>GAT: Two <u>giraffes</u> standing <u>next to each other</u> in a field. Base: Two giraffes standing in a field. AoANet: Two giraffes standing next to each other in a field. GT1: A couple of giraffe standing next to each other. GT2: Two giraffes are facing in opposite directions with their necks bent down. GT3: A couple of giraffes that are standing in the grass.</p>
	<p>GAT: Two <u>motorcycles</u> parked <u>in front of</u> a brick <u>building</u>. Base: Two motorcycles parked next to a building. AoANet: Two motorcycles parked in front of a building. GT1: Two police motorcycles are parked outside a brick building. GT2: Two NYPD motorcycles are parked in front of a building. GT3: Two NYPD motorcycles parked on the street in front of a brick building.</p>		<p>GAT: A <u>bench</u> sitting <u>under</u> a <u>bridge</u> <u>next to</u> a river. Base: A couple of chairs and a bathing in a park. AoANet: A bench sitting next to a body of water. GT1: A bench sitting along side of river next to tree. GT2: A park bench near a bridge and some water. GT3: A bench is out near a body of water</p>
	<p>GAT: A <u>desk</u> with two <u>computer monitors</u> on <u>top of</u> it. Base: A desk with two computer monitors. AoANet: A desk with two computer monitors and a keyboard. GT1: A white table topped with two desktop monitors. GT2: A computer with two monitors on the desk. GT3: A desk with computer and office equipment on it.</p>		<p>GAT: A <u>woman and a girl</u> standing <u>under</u> an <u>umbrella</u>. Base: A man and a woman holding an umbrella. AoANet: A woman and a woman holding an umbrella. GT1: A mother and daughter laugh under umbrellas on a rainy day. GT2: A woman with a black umbrella and a child with a red and black umbrella. GT3: A woman and a little girl holding their umbrellas.</p>
	<p>GAT: Two cell <u>phones</u> sitting <u>next to each other</u> on <u>top of</u> a <u>table</u>. Base: Two cell phones sitting on a table. AoANet: Two cell phones sitting on top of a table. GT1: A pink phone is sitting next to a white phone on a table. GT2: Two types of devices sitting next to each other on a wook table. GT3: An old style phone next to a newer phone.</p>		<p>GAT: A white wedding <u>cake</u> sitting in the grass <u>under</u> a <u>bridge</u>. Base: A white cake sitting in the grass in a field. AoANet: A white wedding cake sitting in the middle of a field. GT1: A white frosted cake sitting in front of some white flowers. GT2: A cake laying on the ground near flowers. GT3: A white cake is by a bunch of flower.</p>
	<p>GAT: A <u>stuffed animal</u> sitting <u>in front of</u> a bowl of <u>oranges</u> on a <u>table</u>. Base: A stuffed animal sitting on a table with a UNK. AoANet: A stuffed teddy bear sitting on a table with a bowl of oranges. GT1: A monkey stuffed toy sitting on a table in front of a bowl of oranges. GT2: A table with a bowl of oranges on top. GT3: A stuffed animal sitting on a table, with oranges behind it.</p>		<p>GAT: A birthday <u>cake</u> with a <u>knife</u> <u>next to it</u> on a <u>table</u>. Base: A birthday cake with a knife on a table. AoANet: A birthday cake with a knife on top of it. GT1: A large cake and a knife on a table. GT2: A grey table with a white and red cake next to knife. GT3: A large red and white sheet cake sitting on top of a table.</p>

Fig. 3. The output comparison of captions generated respectively by our GAT and a base model, as well as three manual description captions of ground truth, GT1~GT3.





	<p>GAT: A colorful boat sitting in the water. GT1: People walk by a boat sitting on the water. GT2: An elaborate boat reflects against the water below it. GT3: A colorful canoe coasting the lake by many people.</p>		<p>GAT: A dog wearing a hat holding a cell phone. GT1: A brown dog earing a yellow hat while sitting on a person's lap. GT2: A dog wearing a bonet on her head. GT3: A dog with a hat strapped to its head.</p>
	<p>GAT: A market with lots of fruits and vegetables on it. GT1: A woman in a pink shirt at a farmers market beside a road. GT2: A road side market filled with lots of watermelon and other things. GT3: Fruit and vegetable stand with lots of watermelons.</p>		<p>GAT: Two street signs on top of a street sign. GT1: A street sign named after a professional athlete. GT2: A couple of green street signs on a pole. GT3: Some street signs on the corner of to joining streets.</p>

Fig. 4. Some typical image captioning results with wrong words, generated by our GAT under non-optimal parameter training.

captioning performance. In addition, the experimental comparisons (offline & online) also show that our GAT framework could often outperform other state-of-the-art ones on both MS COCO and Flickr30k.

In the future, besides geometry relationships, some typical topics on latent semantic information, such as object action and intention, could be worthy of further research. In terms of neural network frameworks,

the *Vision Transformers* are currently receiving more and more research attention. Therefore, some highly-efficient neural frameworks like the *Vision Transformer* and more reliable machine learning algorithms would be believed to be the hot spots of future research in image captioning.

CRediT authorship contribution statement

Chi Wang: Methodology, Conceptualization, Software, Investigation, Writing – original draft. **Yulin Shen:** Investigation, Data curation, Formal analysis, Writing – review & editing. **Luping Ji:** Validation, Writing – review & editing, Supervision, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (NSFC) under Grant No. 61972072 and partly by Sichuan Science and Technology Program under Grant No. 2021YFS0071. We would like to thank the Editors and Reviewers for their valuable revision suggestions.

Appendix. The computation of metrics

BLEU. Bilingual Evaluation Understudy (BLEU) (Papineni et al., 2002) is the first metric to measure the quality of machine generated text. A set of n consecutive words is called an n -gram. BLEU scores measure how many n -grams of generated text appear in reference texts, which is also known as “precision”.

$$P_n(c, S) = \frac{\sum_k \min(h_k(c), \max_{j \in m} (h_k(S_j)))}{\sum_k h_k(c)} \quad (A.1)$$

c is the generated text, S is the set of reference text, and S_j is the j th reference text. $h_k(s)$ indicates the number of k th n -gram of c appears in text s .

$$\text{BLEU} = b_p \cdot \exp\left(\sum_{n=1}^N w_n \log P_n\right) \quad (A.2)$$

w_n is the weight assign to different n -gram scores P_n , which is usually set to be $1/N$. b_p is a penalty factor to force the length of the generated text no shorter than the reference text. $b_p = \exp(1 - l_c/l_s)$, when $l < c$, otherwise $b_p = 1$, where l_s is the length of the shortest reference text, l_c is the length of the generated text.

ROUGE. Recall-Oriented Understudy for Gisting Evaluation (ROUGE) (Lin, 2004) is a set of metrics that are used for measuring the quality of text summary, and it mainly focuses on the recall of n -grams of the reference text.

$$\text{ROUGE-N}(c, S) = \frac{\sum_{j=1}^{|S|} \sum_k \min(h_k(c), h_k(S_j))}{\sum_{j=1}^{|S|} M_j} \quad (A.3)$$

M_j is the number of n -grams in S_j .

There are many variants of ROUGE, such as ROUGE-S and ROUGE-L which takes longest common subsequence problem into account sentence level structure similarity and identifies longest co-occurring in sequence n -grams.

METEOR. Metric for Evaluation of Translation with Explicit Ordering (METEOR) (Banerjee & Lavie, 2005) is a metric calculated by the harmonic mean of unigram precision and recall, where recall plays a more important role than precision. Besides exact word matching,

METEOR has stemming and synonymy matching, which leads to a better correlation with human judgements than BLEU.

The unigrams in the generated text are mapped to those in reference texts by the matching of exact word, stemming and synonymy. METEOR scores could be calculated by following equations.

$$P_m = \frac{m}{\sum_k h_k(c)} \quad (A.4)$$

$$R_m = \frac{m}{\sum_k h_k(S_j)} \quad (A.5)$$

$$F_{mean} = \frac{P_m R_m}{\alpha P_m + (1 - \alpha) R_m} \quad (A.6)$$

$$p = \gamma \left(\frac{ch}{m}\right)^\theta \quad (A.7)$$

$$M = (1 - p) F_{mean} \quad (A.8)$$

where m is the number of unigrams in the candidate translation which appear in the reference translations. $h_k(s)$ indicates the number of k th unigram appears in text s . So P_m is the precision metric and R_m is the recall metric. F_{mean} is the harmonic mean of precision and recall. In order to take longer sentences into account rather than unigrams, the penalty p is imported. Unigrams are grouped into the fewest possible chunks, where a chunk is defined as a set of unigrams that are adjacent in the generated text and in the reference text. The longer the adjacent mappings between the candidate and the reference, the fewer chunks there are. So ch is the number of the chunks and M is the METEOR score. α, γ, θ are usually set to 0.1, 0.5, 3 for fair comparisons.

CIDEr. Consensus-based Image Description Evaluation (CIDEr) (Vedantam et al., 2015) is an automatic consensus metric for evaluating image descriptions. For n -grams, CIDEr_n could be gained from:

$$\text{CIDEr}_n(c, S) = \frac{1}{|S|} \sum_{j=1}^{|S|} \frac{\mathbf{g}^n(c) \cdot \mathbf{g}^n(S_j)}{\|\mathbf{g}^n(c)\| \cdot \|\mathbf{g}^n(S_j)\|} \quad (A.9)$$

where c is the generated text, and S is the set of reference text, and $\mathbf{g}^n(s)$ is the vector formed by TF-IDF scores of all n -grams in text s .

TF-IDF is short for Term Frequency-Inverse Document Frequency, which measures the importance of a word in a specific sentence. The more the word appears in the text, and the less the word appears in the whole corpus, the more important it would be.

$$g_k(c) = \text{TF}(k) * \text{IDF}(k) \quad (A.10)$$

$$\text{TF}(k) = \frac{h_k(c)}{\sum_l h_l(c)} \quad (A.11)$$

$$\text{IDF}(k) = \log\left(\frac{N}{\sum_{i=1}^N \min(1, \sum_{j=1}^{|S^i|} h_k(S_j^i))}\right) \quad (A.12)$$

The TF-IDF score of the k th n -gram could be gained from $g_k(c)$. N is the number of all cases, each of which consists of the generated sentence and a set of reference sentences. S^i is the set of reference sentences of i th case, and S_j^i means the j th sentence in S^i . The scores of all n -grams in c form the $\mathbf{g}^n(c)$ vector.

And the overall CIDEr score is the average of the CIDEr_n set:

$$\text{CIDEr} = \frac{1}{N} \sum_{n=1}^N \text{CIDEr}_n(c, S) \quad (A.13)$$

When optimizing an algorithm for a specific metric undesirable results may be achieved. The gaming of a metric may result in sentences with high scores, yet produce poor results when judged by a human. CIDEr-D is a variant of CIDEr, and is more robust to “gaming”.

$$\text{CIDEr-D}_n(c, S) = \frac{10}{|S|} \sum_j e^{\frac{-(l(c)-l(S_j))^2}{2\sigma^2}} * \frac{\min(\mathbf{g}^n(c), \mathbf{g}^n(S_j)) \cdot \mathbf{g}^n(S_j)}{\|\mathbf{g}^n(c)\| \cdot \|\mathbf{g}^n(S_j)\|} \quad (A.14)$$

$l(c)$ is the length of the generated sentence, and $l(S_j)$ is the length of the reference sentence S_j .

SPICE. Semantic Propositional Image Caption Evaluation (SPICE) (Anderson et al., 2016) is a new caption evaluation metric based on

semantic concept. SPICE uses a graph-based semantic representation to encode the objects, attributes, and relationships in captions. It first parses the to-be-evaluated captions and reference captions into syntactic dependency trees using Probabilistic Context-Free Grammar (PCFG) dependency parser, and then maps the dependency trees into scene graphs using a rule-based approach. Finally, it calculates the F-score value of the objects, attributes and relationships in the caption to be evaluated.

$$\begin{aligned} \text{SPICE}(c, S) &= F_1(c, S) \\ &= \frac{2 \cdot P(c, S) \cdot R(c, S)}{P(c, S) + R(c, S)} \end{aligned} \quad (\text{A.15})$$

$$P(c, S) = \frac{|T(G(c)) \otimes T(G(S))|}{|T(G(c))|} \quad (\text{A.16})$$

$$R(c, S) = \frac{|T(G(c)) \otimes T(G(S))|}{|T(G(S))|} \quad (\text{A.17})$$

$G(\cdot)$ means the function to convert a sentence to its corresponding graph scene, and $T(\cdot)$ is the function to convert a graph scene to a series of tuples. \otimes is similar to the intersection while it allows synonymy matching.

References

- Anderson, P., Fernando, B., Johnson, M., & Gould, S. (2016). SPICE: Semantic propositional image caption evaluation. In *Proceedings of the 14th European conference on computer vision, ECCV 2016* (pp. 382–398).
- Anderson, P., He, X., Buehler, C., Teney, D., Johnson, M., Gould, S., et al. (2018). Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of 2018 IEEE/CVF conference on computer vision and pattern recognition (CVPR)* (pp. 6077–6086). <http://dx.doi.org/10.1109/CVPR.2018.00636>.
- Bai, S., & An, S. (2018). A survey on automatic image caption generation. *Neurocomputing*, 311, 291–304.
- Banerjee, S., & Lavie, A. (2005). [METEOR]: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the Acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization* (pp. 65–72).
- Cai, W., & Liu, Q. (2020). Image captioning with semantic-enhanced features and extremely hard negative examples. *Neurocomputing*, 413, 31–40. <http://dx.doi.org/10.1016/j.neucom.2020.06.112>.
- Chen, X., Jiang, M., & Zhao, Q. (2021). Self-distillation for few-shot image captioning. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision* (pp. 545–555).
- Chen, H., Wang, Y., Yang, X., & Li, J. (2021). Captioning transformer with scene graph guiding. In *2021 IEEE international conference on image processing (ICIP)* (pp. 2538–2542). <http://dx.doi.org/10.1109/ICIP42928.2021.9506193>.
- Cornia, M., Stefanini, M., Baraldi, L., & Cucchiara, R. (2020). Meshed-memory transformer for image captioning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 10578–10587).
- Farhadi, A., Hejrati, M., Sadeghi, M. A., Young, P., Rashtchian, C., Hockenmaier, J., et al. (2010). Every picture tells a story: Generating sentences from images. In *European conference on computer vision* (pp. 15–29). Springer.
- Gao, L., Fan, K., Song, J., Liu, X., Xu, X., & Shen, H. T. (2019). Deliberate attention networks for image captioning. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 33 (pp. 8320–8327).
- Guo, L., Liu, J., Zhu, X., Yao, P., Lu, S., & Lu, H. (2020). Normalized and geometry-aware self-attention network for image captioning. In *2020 IEEE/CVF conference on computer vision and pattern recognition (CVPR)* (pp. 10324–10333). <http://dx.doi.org/10.1109/CVPR42600.2020.01034>.
- Gupta, A., Verma, Y., & Jawahar, C. V. (2012). Choosing linguistics over vision to describe images. In *In twenty-sixth national conference on artificial intelligence* (pp. 606–612).
- Herdade, S., Kappeler, A., Boakye, K., & Soares, J. (2019). Image captioning: Transforming objects into words. In *Advances in neural information processing systems* (pp. 11135–11145).
- Huang, L., Wang, W., Chen, J., & Wei, X.-Y. (2019). Attention on attention for image captioning. In *Proceedings of the IEEE international conference on computer vision* (pp. 4634–4643).
- Jiang, W., Ma, L., Jiang, Y.-G., Liu, W., & Zhang, T. (2018). Recurrent fusion network for image captioning. In *Proceedings of the European conference on computer vision (ECCV)* (pp. 499–515).
- Karpathy, A., & Fei-Fei, L. (2015). Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3128–3137).
- Karpathy, A., & Fei-Fei, L. (2017). Deep visual-semantic alignments for generating image descriptions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(4), 664–676. <http://dx.doi.org/10.1109/TPAMI.2016.2598339>.
- Karpathy, A., Joulin, A., & Li, F. F. (2014). Deep fragment embeddings for bidirectional image sentence mapping. In *Proceedings of the twenty seventh advances in neural information processing systems (NIPS)*, Vol. 3 (pp. 1889–1897).
- Krishna, R., Zhu, Y., Groth, O., Johnson, J., Hata, K., Kravitz, J., et al. (2017). Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123(1), 32–73.
- Kulkarni, G., Premraj, V., Ordonez, V., Dhar, S., Li, S., Choi, Y., et al. (2013). Babytalk: Understanding and generating simple image descriptions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(12), 2891–2903. <http://dx.doi.org/10.1109/TPAMI.2012.162>.
- Li, G., Zhu, L., Liu, P., & Yang, Y. (2019). Entangled transformer for image captioning. In *Proceedings of the IEEE international conference on computer vision* (pp. 8928–8937).
- Lin, C.-Y. (2004). ROUGE: A package for automatic evaluation of summaries. In *Text summarization branches out* (pp. 74–81). Barcelona, Spain: Association for Computational Linguistics.
- Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., et al. (2014). Microsoft COCO: Common objects in context. In *European conference on computer vision* (pp. 740–755). Springer.
- Liu, S., Ren, Z., & Yuan, J. (2021). SibNet: Sibling convolutional encoder for video captioning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(9), 3259–3272. <http://dx.doi.org/10.1109/TPAMI.2019.2940007>.
- Lu, Y., Guo, C., Dai, X., & Wang, F.-Y. (2022). Data-efficient image captioning of fine art paintings via virtual-real semantic alignment training. *Neurocomputing*, 490, 163–180. <http://dx.doi.org/10.1016/j.neucom.2022.01.068>.
- Lu, J., Xiong, C., Parikh, D., & Socher, R. (2017). Knowing when to look: Adaptive attention via a visual sentinel for image captioning. In *The IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 3242–3250). <http://dx.doi.org/10.1109/CVPR.2017.345>.
- Ma, L., Lu, Z., Shang, L., & Li, H. (2015). Multimodal convolutional neural networks for matching image and sentence. In *Proceedings of 2015 IEEE international conference on computer vision* (pp. 2623–2631).
- Mao, J., Wei, X., Yang, Y., Wang, J., Huang, Z., & Yuille, A. L. (2015). Learning like a child: Fast novel visual concept learning from sentence descriptions of images. In *Proceedings of 2015 IEEE international conference on computer vision (ICCV)* (pp. 2533–2541). <http://dx.doi.org/10.1109/ICCV.2015.291>.
- Mao, J., Xu, W., Yang, Y., Wang, J., Huang, Z., & Yuille, A. (2014). Deep captioning with multimodal recurrent neural networks (m-RNN). arXiv preprint [arXiv:1412.6632](https://arxiv.org/abs/1412.6632).
- Nabati, M., & Behrad, A. (2021). Multimodal video-text matching using a deep bifurcation network and joint embedding of visual and textual features. *Expert Systems with Applications*, 184, 115541. <http://dx.doi.org/10.1016/j.eswa.2021.115541>.
- Ordonez, V., Kulkarni, G., & Berg, T. L. (2011). Im2Text: Describing images using 1 million captioned photographs. In *Proceedings of the advances in neural information processing systems (NIPS)* (pp. 1143–1151).
- Oruganti, R. M., Sah, S., Pillai, S., & Ptucha, R. (2016). Image description through fusion based recurrent multi-modal learning. In *Proceedings of 2016 IEEE international conference on image processing (ICIP)* (pp. 3613–3617). <http://dx.doi.org/10.1109/ICIP.2016.7533033>.
- Papineni, K., Roukos, S., Ward, T., & Zhu, W.-J. (2002). [Bleu]: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics* (pp. 311–318). Association for Computational Linguistics.
- Rennie, S. J., Marcheret, E., Mroueh, Y., Ross, J., & Goel, V. (2017). Self-critical sequence training for image captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 7008–7024).
- Shen, L., & Wang, Y. (2022). TCCT: Tightly-coupled convolutional transformer on time series forecasting. *Neurocomputing*, 480, 131–145. <http://dx.doi.org/10.1016/j.neucom.2022.01.039>.
- Socher, R., & Fei-Fei, L. (2010). Connecting modalities: Semi-supervised segmentation and annotation of images using unaligned text corpora. In *Proceedings of 2010 IEEE computer society conference on computer vision and pattern recognition* (pp. 966–973). IEEE.
- Ushiku, Y., Yamaguchi, M., Mukuta, Y., & Harada, T. (2015). Common subspace for model and similarity: Phrase learning for caption generation from images. In *Proceedings of 2015 IEEE international conference on computer vision (ICCV)* (pp. 2668–2676). <http://dx.doi.org/10.1109/ICCV.2015.306>.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., et al. (2017). Attention is all you need. In *Advances in neural information processing systems* (pp. 5998–6008).
- Vedantam, R., Lawrence Zitnick, C., & Parikh, D. (2015). CIDEr: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4566–4575).
- Vinyals, O., Toshev, A., Bengio, S., & Erhan, D. (2015). Show and tell: A neural image caption generator. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3156–3164).
- Xian, T., Li, Z., Zhang, C., & Ma, H. (2022). Dual global enhanced transformer for image captioning. *Neural Networks*, 148, 129–141. <http://dx.doi.org/10.1016/j.neunet.2022.01.011>.

- Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., et al. (2015). Show, attend and tell: Neural image caption generation with visual attention. In *The 32nd international conference on machine learning* (pp. 2048–2057).
- Yan, C., Hao, Y., Li, L., Yin, J., Liu, A., Mao, Z., et al. (2022). Task-adaptive attention for image captioning. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(1), 43–51. <http://dx.doi.org/10.1109/TCSVT.2021.3067449>.
- Yan, F., & Mikolajczyk, K. (2015). Deep correlation for matching images and text. In *Proceedings of 28th IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 3441–3450). <http://dx.doi.org/10.1109/CVPR.2015.7298966>.
- Yang, X., Tang, K., Zhang, H., & Cai, J. (2019). Auto-encoding scene graphs for image captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 10685–10694).
- Yao, T., Pan, Y., Li, Y., & Mei, T. (2018). Exploring visual relationship for image captioning. In *Proceedings of the European conference on computer vision (ECCV)* (pp. 684–699).
- You, Q., Jin, H., Wang, Z., Fang, C., & Luo, J. (2016). Image captioning with semantic attention. In *Proceedings of 2016 IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 4651–4659). <http://dx.doi.org/10.1109/CVPR.2016.503>.
- Young, P., Lai, A., Hodosh, M., & Hockenmaier, J. (2014). From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2, 67–78.
- Yu, L., Zhang, J., & Wu, Q. (2022). Dual attention on pyramid feature maps for image captioning. *IEEE Transactions on Multimedia*, 24, 1775–1786. <http://dx.doi.org/10.1109/TMM.2021.3072479>.
- Zhang, J., Li, K., Wang, Z., Zhao, X., & Wang, Z. (2021). Visual enhanced gLSTM for image captioning. *Expert Systems with Applications*, 184, 115462. <http://dx.doi.org/10.1016/j.eswa.2021.115462>.
- Zhang, X., Sun, X., Luo, Y., Ji, J., Zhou, Y., Wu, Y., et al. (2021). RSTNet: Captioning with adaptive attention on visual and non-visual words. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 15465–15474). <http://dx.doi.org/10.1109/CVPR46437.2021.01521>.
- Zhang, M., Yang, Y., Chen, X., Ji, Y., Xu, X., Li, J., et al. (2021). Multi-stage aggregated transformer network for temporal language localization in videos. In *2021 IEEE/CVF conference on computer vision and pattern recognition (CVPR)* (pp. 12664–12673). <http://dx.doi.org/10.1109/CVPR46437.2021.01248>.
- Zhou, L., Palangi, H., Zhang, L., Hu, H., Corso, J. J., & Gao, J. (2020). Unified vision-language pre-training for image captioning and VQA. In *Proceedings of the thirty-fourth aaai conference on artificial intelligence* (pp. 13041–13049).
- Zhu, X., Li, L., Liu, J., Peng, H., & Niu, X. (2018). Captioning transformer with stacked attention modules. *Applied Sciences*, 8(5), 739.