



# Dual Global Enhanced Transformer for image captioning

Tiantao Xian<sup>a</sup>, Zhixin Li<sup>a,\*</sup>, Canlong Zhang<sup>a</sup>, Huifang Ma<sup>b</sup>

<sup>a</sup> Guangxi Key Lab of Multi-source Information Mining and Security, Guangxi Normal University, Guilin 541004, China

<sup>b</sup> College of Computer Science and Engineering, Northwest Normal University, Lanzhou 730070, China

## ARTICLE INFO

### Article history:

Received 4 August 2021

Received in revised form 17 November 2021

Accepted 16 January 2022

Available online 21 January 2022

### Keywords:

Image captioning

Transformer

Global information

Visual attention

Reinforcement learning

## ABSTRACT

Transformer-based architectures have shown great success in image captioning, where self-attention module can model source and target interaction (e.g., object-to-object, object-to-word, word-to-word). However, the global information is not explicitly considered in the attention weight calculation, which is essential to understand the scene content. In this paper, we propose Dual Global Enhanced Transformer (DGET) to incorporate global information in the encoding and decoding stages. Concretely, in DGET, we regard the grid feature as the visual global information and adaptively fuse it into region features in each layer by a novel Global Enhanced Encoder (GEE). During decoding, we proposed Global Enhanced Decoder (GED) to explicitly utilize the textual global information. First, we devise the context encoder to encode the existing caption generated by classic captioner as a context vector. Then, we use the context vector to guide the decoder to generate accurate words at each time step. To validate our model, we conduct extensive experiments on the MS COCO image captioning dataset and achieve superior performance over many state-of-the-art methods.

© 2022 Elsevier Ltd. All rights reserved.

## 1. Introduction

Image captioning aims to describe the semantic content of an image in natural language, which has recently attracted extensive research attention. In the past few years, most image captioning models (Anderson et al., 2018; Huang, Wang, Chen, & Wei, 2019; Vinyals, Toshev, Bengio, & Erhan, 2015; Xu et al., 2015) are based on encoder–decoder framework, where the encoder adopts Convolution Neural Networks (CNN) to encode an image into a feature vector, and the decoder generates sentence word by word using Recurrent Neural Network (RNN). Attention mechanism is applied to prompt the decoder solely focus on the crucial region of image, and bring significant improvements on most evaluation metrics. With the success of transformer architecture (Vaswani et al., 2017) in natural language processing, many recent works have studied its application in vision-language tasks (Cornia, Stefanini, Baraldi, & Cucchiara, 2020; Ji et al., 2021; Li, Zhu, Liu, & Yang, 2019), where an amount of self-attention module is utilized to capture the correlations among intra- and inter-modal.

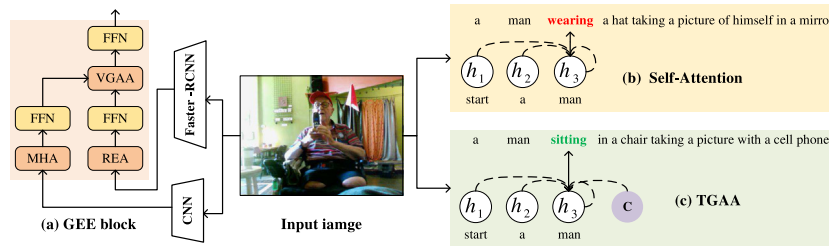
Despite the success that existing Transformer-based approaches have achieved, they have the following limitations: (1) In the encoding stage, the region features may fail to cover all objects in the image, although the attention mechanism models the region-level relationship, they still lack guidance from image-level information. It causes the problem of small object missing

and incorrect relationship recognition between objects; (2) In the decoding stage, the prediction word of the decoder in the current time step only depends on the previously generated word. In this case, the decoder is unable to capture sufficient semantic information, especially at the early time step, resulting in the generation of inaccurate words and then affecting the quality of the generated word at the later time step. Fig. 1 shows our Global Enhanced Encoder block, and the difference between traditional self-attention and our textual global adaptive attention. As illustrated in Fig. 1(b), the self-attention of subsequences in autoregressive decoder can only capture the previously generated semantic information, i.e., “a man”, which is common in datasets. Therefore, decoder tends to predict “wearing hat” by experiences, which is far from the semantic content of input image.

To solve the above problems, we propose Dual Global Enhanced Transformer (DGET), which consists of Global Enhanced Encoder (GEE) and Global Enhanced Decoder (GED). GEE and GED can capture the visual global information in the encoding stage, and perceive the textual global information in the decoding stage. In the coding phase, GEE enables the complementary advantages of region-level and grid-level for a better visual representation. As shown in Fig. 1(a), we first explore the intra-relationship of the two source visual features separately through two independent self-attention modules, where the relative geometry information of region feature was considered via Relation Enhanced Attention (REA). After that, the grid features are integrated into the region features by Visual Global Adaptively Attention (VGAA) module. The intuition is that the grid features contain more fine-grained

\* Corresponding author.

E-mail address: [lizx@gxnu.edu.cn](mailto:lizx@gxnu.edu.cn) (Z. Li).



**Fig. 1.** The Core modules of our approach, and comparison with traditional methods. (a) is our Global Enhanced Encoder Block. Comparison between traditional self-attention (b) and our textual global adaptive attention (c). The traditional self-attention depends solely on the previously generated words when predicting word at current time step. Our method can take advantage of the textual global information to guide model learning at each time step.

information, while the region features contain object information. They can complement each other to achieve a better visual representation, which is essential to improve captioning performance. In GED, we first design a Context Encoder to encode the existing caption as context vector. Subsequently, we fuse the context vector into the decoder at each time step via Textual Global Adaptive Attention (TGAA), as illustrated in Fig. 1(c). The intuition is that when humans understand a scene, they first find an approximate understanding based on experience, and then fine-tune it. On the other hand, the existing caption can serve as prior knowledge to guide the decoder to generate a more accurate caption. Hence, we regard the existing captions generated by the classical image captioning model as a rough understanding of the scene, and then incorporate it into decoder to mimic this process.

We conduct extensive experiments and analyses on the challenging Microsoft COCO image captioning benchmark to evaluate our proposed method. To validate the adaptability of our method, we explored three different context encoders (LSTM, BiLSTM and Transformer) as well as two popular image captioners (Up-Down (2018) and AoANet (2019)) and achieved significant improvements compared with strong baselines.

To sum up, our major contributions are itemized below:

- We propose Global Enhanced Encoder (GEE) to refine the representation of region features by leveraging grid features to provide more detailed information. At the same time, geometric information was combined to model complex spatial relationships of region features.
- We design a Global Enhanced Decoder (GED), the decoder can adaptively attend the textual global information at each time step, which is crucial for learning complete semantic information. In addition, we devise three different variants of Context Encoder to explore the impact of different representations of the existing caption on performance.
- Extensive experiments on the benchmark MS COCO image captioning dataset are conducted to quantitatively and qualitatively prove the usefulness of the proposed models. The experimental results demonstrate that our proposed method performs much better than other state-of-the-art methods.

## 2. Related works

### 2.1. Image captioning

With the development of deep learning, image captioning methods based on deep neural network has been widely studied with superior performance. Existing image captioning approaches typically follow the encoder–decoder paradigm (Huang, Li, Wei, Zhang, & Ma, 2020; Qin, Du, Zhang, & Lu, 2019; Sammani & Melas-Kyriazi, 2020; Wei, Li, Huang, Zhang, Ma, & Shi, 2021; Wei, Li, Zhang, & Ma, 2020) that firstly utilizes convolution neural network (CNN) to encode image and then adopts recurrent neural

network (RNN) based decoder to generate the sentence word by word. To mimic the human ability to describe different words in different regions, attention mechanism is introduced into this task. Xu et al. (2015) proposed the soft and hard attention to automatically focus on the different regions according to current hidden states. Lu, Xiong, Parikh, and Socher (2017) improved this work through introducing an adaptively attention mechanism, enabling models to decide whether to attend to the image and where. Chen et al. (2017) extend the attention mechanism to channel-wise and multi-layer of CNN feature map to better explore where (i.e., spatial) and what (i.e., channel-wise) of attention during sentence generation. Most of previous works use the feature from last layer of Convolution layer as visual input, i.e.,  $N \times N$  grid-level feature. To further improve the representation of image, Anderson et al. (2018) take pre-trained Faster-RCNN (Ren, He, Girshick, & Sun, 2016) on Visual Genome dataset (Krishna et al., 2017) to extract region-level features, enables attention to be calculated more similar to human at the level of objects. Yao, Pan, Li, and Mei (2019) proposed Hierarchical Attention to use patch, object and text features simultaneously, which enable the model generating different words according to different features. Recently, a variety of advanced models attempt to incorporate high-level semantic information for boost image captioning (Fang et al., 2015; Li, Zhu, et al., 2019; Liu, Liu, Ren, He, & Sun, 2019; Wu, Shen, Liu, Dick, & Van Den Hengel, 2016; Yao, Pan, Li, Qiu, & Mei, 2017). You, Jin, Wang, Fang, and Luo (2016) propose a semantic attention which exploits not only an overview understanding of input image, but also abundant fine-grain visual semantic aspects. Wang, Bai, Zhang, and Lu (2020) propose recall mechanism for image captioning, which recalls a set of word from corpus via text-retrieval module and incorporates the recall words by attention and gate mechanism. In addition, some new state-of-the-art works explore using scene graph to enhance the semantic representation of image. The scene graph contains the structured semantic information of the input image, which includes the knowledge of objects, attributes, and relationships. Yao, Pan, Li, and Mei (2018) proposed the GCN-LSTM architecture, modeling semantic and spatial object relationship to enrich region-level representations by graph convolutional networks. Yang, Tang, Zhang, and Cai (2019) proposed the SGAE model, encoding the inductive bias into a dictionary to help the model alleviate over fitting to the datasets bias and improve reasoning ability.

### 2.2. Transformer based image captioning

With the success of Transformer architecture (Vaswani et al., 2017) in the field of natural language processing, some recent works (Guo, Liu, Zhu, Yao, Lu, & Lu, 2020; Liu, Wu, Ge, Zhang, Fan, & Zou, 2020; Pan, Yao, Li, & Mei, 2020; Yu, Li, Yu, & Huang, 2019) use it to replace RNN as decoder, achieving new state-of-the-art performance. Li, Zhu, et al. (2019) propose Entangled Transformer to exploit complementary information of visual region and semantic attributes simultaneously. Herdade, Kappeler,

Boakye, and Soares (2019) modify the calculation of self-attention module to not only solely depend on pairwise similarity, but also consider the geometry relationship between objects. Cornia et al. (2020) propose Mesh-Memory Transformer to exploit a prior knowledge through learning memory vectors and introduce a novel meshed connectivity between encoding and decoding modules. Yu et al. (2019) proposed Multimodal Transformer model, which views image features obtained from different visual feature extractors as different views, and simultaneously captures intra- and inter-modal interactions in the self-attention module.

More similar to our work, Ji et al. (2021) present Global Enhanced Transformer (GET) to capture both intra- and inter-level visual global representation. Specifically, an additional global vector is learned at each layer of the encoder (i.e., the intra-layer global representation). Then, an LSTM is used to refine the global representation from multiple layers to obtain the inter-layer global representation. The output of the encoder and the global representation are concatenated to enable the decoder to capture both local and global visual information in the cross-modal attention module at each time step.

Although the aforementioned methods have achieved advanced performance, it still lacks consideration of the textual global information. Moreover, leveraging visual and textual global information simultaneously and exploiting visual features from different sources for the image captioning task has never been explored, which motivates our work in this paper.

### 2.3. Vision and language pre-training

Several works (Chen et al., 2020; Li, Duan, Fang, Gong, & Jiang, 2020; Li, Yatskar, Yin, Hsieh, & Chang, 2019; Lu, Batra, Parikh, & Lee, 2019; Tan & Bansal, 2019) are devoted to vision-language pre-training (VLP) on large-scale multi-modal datasets, learning generic representations and then transferring them to downstream tasks for fine-tuning. These VLP models are based on multi-layer transformers, and we categorize them into two categories: single-stream Transformers and two-stream Transformers. The single-stream design feed multi-modal inputs to a single Transformer, while the two-stream design first feeds data into a modality-independent Transformers and then uses another Transformers to learn the cross-modal representation. These VLP methods have two characteristics. First, in order not to expose downstream tasks, the pre-training task for these methods is task-agnostic, such as Masked Language Modeling (MLM), Image-Text Matching (ITM), and Masked Region Classification (MRC). Second, to learn the generic representation, larger batch sizes and millions of training data are required compared to non-pre-trained methods. For this purpose, using batch sizes of 512 or larger, and aggregating multiple multi-modal datasets as training sets are common strategies. For example, Li, Yin, et al. (2020) built the pre-training corpus based on the existing V+L datasets, including MS COCO (Lin et al., 2014), Conceptual Captions (Sharma, Ding, Goodman, & Soricut, 2018), SBU captions (Ordóñez, Kulka-rni, & Berg, 2011), flicker30k (Young, Lai, Hodosh, & Hockenmaier, 2014), GQA (Hudson & Manning, 2019) etc. In total, the corpus consists of 6.5 million text-tag-image triples, and using batch sizes of 768.

Despite the advanced performance achieved by the above methods, it is difficult to collect large multi-modal data in the real world and the demand on computational resources increases the application cost. On the contrary, our method is non-pre-trained and focuses on training from scratch based on MSCOCO datasets containing 0.55 million image-text pairs, which is one-tenth of the amount of data required to train a VLP model. As a result, our training is faster and less demanding on computational resources.

## 3. Proposed method

In this section, we first briefly describe the preliminary knowledge of the plain Transformer architecture. Then, we introduce our Dual Global Enhanced Transformer architecture for image captioning, which consists of Global Enhanced Encoder (GEE), Global Enhanced Decoder (GED), Classic Captioner and Context Encoder. In the encoding stage, the GEE explicitly utilizes the grid-level features to enhance the representation of region-level features. During decoding, we first obtain a coarse caption and encode it as a context vector by the Classic Captioner and Context Encoder. Then, the GED adaptively incorporates the context vector to generate captions word by word. The overall structure of our model is shown in Fig. 3.

### 3.1. Transformer model

The Transformer model (Vaswani et al., 2017) was first proposed for machine translation, and has been successfully applied to many natural language processing tasks. Transformer consists of a set of multi-head self-attention(MHA) module and feed-forward network(FFN) module. We first introduce the MHA which is the core component of the Transformer. The input of the MHA consists of a query  $q \in \mathbb{R}^d$ , keys  $k_t \in \mathbb{R}^d$  and values  $v_t \in \mathbb{R}^d$ , where  $t \in \{1, 2, \dots, n\}$  is the number of key-value pairs and  $d$  is the dimensionality of all the inputs features. In order to jointly attend to information from different representation. We linearly project to the queries, keys and values  $h$  times with different, learned linear projections to  $d_h$  dimensions, respectively. Then we concatenate the output of  $h$  head and once again projected, resulting in the final output. In practice, we pack all the keys and values vector into matrices  $K = [k_1, \dots, k_n] \in \mathbb{R}^{n \times d}$  and  $V = [v_1, \dots, v_n] \in \mathbb{R}^{n \times d}$  respectively. For a set of queries  $Q = [q_1, \dots, q_m] \in \mathbb{R}^{m \times d}$ , the attention function can be calculated in parallel as below:

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \dots, \text{head}_h)W^O \quad (1)$$

$$\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$$

Where  $W_i^Q, W_i^K, W_i^V \in \mathbb{R}^{d \times d_h}$  and  $W^O \in \mathbb{R}^{hd_h \times d}$ . In this work, we set  $d_h = d/h$ . Note that, the Attention function is particular “Scaled Dot-Product Attention”, which calculates the dot products of the query with all keys, divide each by  $\sqrt{d_h}$ , and apply a softmax function to obtain the normalized attention weights on the values as follows:

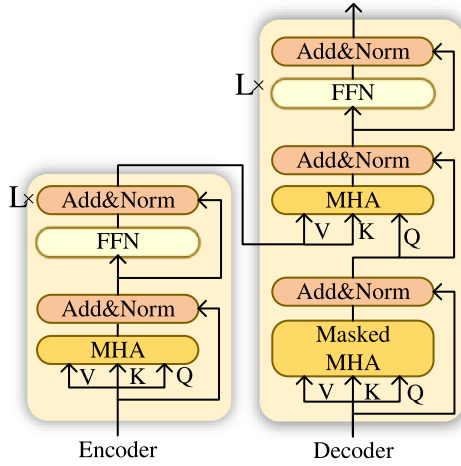
$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_h}}\right)V \quad (2)$$

In addition to MHA module, another basic component in the Transformer is the feed-forward networks (FFN), which follow by MHA in each layer of encoder and decoder. The FFN consists of two fully connected layer with a ReLU activation and dropout function (Srivastava, Hinton, Krizhevsky, Sutskever, & Salakhutdinov, 2014) in between as below:

$$\text{FFN}(x) = \text{FC}(\text{Dropout}(\text{ReLU}(\text{FC}(x)))) \quad (3)$$

where  $x$  is the output of previous MHA module.

The overall architecture of plain Transformer as shown in Fig. 2. Both the encoder and the decoder consist of  $L$  attention block, and each attention block contains the MHA and FFN modules. The MHA module learns the attended features that according to the pairwise similar between query and key vector, and the FFN module further non-linearly transforms the attended features. In the encoder, each attention block is self-attention such that the queries, keys and values in Eq. (2) refer to the same input features. Slightly different from encoder, the attention block



**Fig. 2.** The Transformer architecture, where the encoder and decoder contain  $L$  identical attention blocks respectively. MHA and FFN denote the multi-head attention module and feed-forward network module, respectively.

in the decoder consists of a masked self-attention module and a cross-attention module. It first models the left sub-sequences interaction of given input features by masked self-attention and then models the inter-modal interaction by taking the output features of the encoder as key and value matrix and viewing the output of self-attention module as query vector. Note that since the decoder is not autoregressive, in order not to expose future information during training, we force the current time step to focus only on the left subsequences, i.e., mask the right subsequences. Besides, the residuals connection (He, Zhang, Ren, & Sun, 2016) and layer normalization (Ba, Kiros, & Hinton, 2016) are applied after all the MHA and FFN modules.

### 3.2. Dual global enhanced transformer

As shown in Fig. 3, we devise our Dual Global Enhanced Transformer (DGET) based on the plain Transformer for image captioning. Our DGET consists of Global Enhanced Encoder, Global Enhanced Decoder and Context Encoder, which are introduced in Section 3.2.1, Section 3.2.4 and Section 3.2.5.

#### 3.2.1. Global enhanced encoder

Given an image, we first utilize the bottom-up features  $V_r = \{r_1, r_2, \dots, r_n\}$  provided by Anderson et al. (2018) as region features, where  $r_i \in \mathbb{R}^{2048}$  and  $n$  is the number of region. Moreover, we consider the size and coordinates of the object's bounding box as geometric information, which represented as  $x, y, h, w$  (center coordinates, widths, and heights). We then employed a pre-trained ResNet-152 (He et al., 2016) to obtain grid features  $V_g = \{g_1, g_2, \dots, g_m\}$ , where  $m$  denotes the number of the fragment and  $g_i \in \mathbb{R}^{2048}$ . To adapt the feature dimensionality to the encoder, the visual features  $V_r$  and  $V_g$  are first fed into fully-connected linear layer respectively, then we get projected features  $V_r^0 \in \mathbb{R}^{n \times d}$  and  $V_g^0 \in \mathbb{R}^{m \times d}$ .

We take the calculation process of  $(l + 1)$ -th ( $0 \leq l \leq L$ ) layer of GEE as an example. Visual features from different kinds of feature extractors have different characteristics. We first model intra-level relationships of two kinds of features by MHA module and Relation Enhanced Attention (REA), respectively:

$$O_g^{l+1} = \text{Add\&Norm}(\text{MHA}(v_r^l, v_g^l, v_g^l)) \quad (4)$$

$$O_r^{l+1} = \text{Add\&Norm}(\text{REA}(v_r^l, v_r^l, v_r^l, G^l)) \quad (5)$$

Where the detail of REA and  $G^l$  will be described in Section 3.2.2. The Add&Norm denote the operation of residual connect and layer normalization. Following Xiong et al. (2020), we adopt post layer normalization as below :

$$\text{Add\&Norm}(f(x)) = \text{Dropout}(x + f(\text{LayerNorm}(x))) \quad (6)$$

Then, we employ two independent FFN modules after the MHA module and REA module:

$$V_g^{l+1} = \text{Add\&Norm}(\text{FFN}(O_g^{l+1})) \quad (7)$$

$$M_r^{l+1} = \text{Add\&Norm}(\text{FFN}(O_r^{l+1})) \quad (8)$$

Most transformer-based methods (Cornia et al., 2020; Ji et al., 2021; Li, Zhu, et al., 2019) use region-level visual features as input, since the permutation invariance of self-attention is suited to this kind of feature that is not interdependent with the neighborhood, i.e., changing the order of the regions will not affect their meaning. However, subject to object detector performance, the region-level features may not cover the entire image, which inevitably missing small objects and scenes details. By contrast, these shortcomings are the advantage of grid-level features, which cover all the content of a given image. It is natural that integrate the grid features into the region features to obtain better visual representation. We complete this process via our proposed Visual Global Adaptively Attention (VGAA) module, and we also adopt the FFN module after VGAA:

$$\bar{V}_r^{l+1} = \text{Add\&Norm}(\text{VGAA}(M_r^{l+1}, V_g^{l+1}, V_g^{l+1})) \quad (9)$$

$$V_r^{l+1} = \text{Add\&Norm}(\text{FFN}(\bar{V}_r^{l+1})) \quad (10)$$

Where the detail of VGAA will be described in Section 3.2.3. Fig. 4 shows the calculation flow of the our Relation Enhanced Attention and Visual Global Adaptively Attention modules. After multi-layer encoding, region features are fed into decoder layers.

#### 3.2.2. Relation enhanced attention

Recently, many works based on region features have not explicitly modeled the geometric relationship between objects, which is crucial for the model learning scene content. Following Herdade et al. (2019), we explicitly incorporate the geometric information of the region in the calculation of the self-attention module. The processing flow of Relation Enhanced Attention is shown in Fig. 4(a). Firstly, we calculate the relative position and size of the bounding boxes between two objects  $i$  and  $j$ :

$$\omega(i, j) = \left( \log\left(\frac{|x_i - x_j|}{w_i}\right), \log\left(\frac{|y_i - y_j|}{h_i}\right), \log\left(\frac{w_i}{w_j}\right), \log\left(\frac{h_i}{h_j}\right) \right)^T \quad (11)$$

where  $x_i, y_i, w_i, h_i$  are the center coordinate, width, and height of box  $i$ , respectively. The geometric attention weights are then calculated as:

$$G_{ij} = \text{ReLU}(\text{Emb}(\omega)W_G) \quad (12)$$

Then, we take the effect of  $G$  into the calculation of self-attention as:

$$\text{REA}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d}} + \log(G)\right)V \quad (13)$$

Note that the Emb method is following Vaswani et al. (2017), which embed  $\omega$  to a high-dimensional representation.



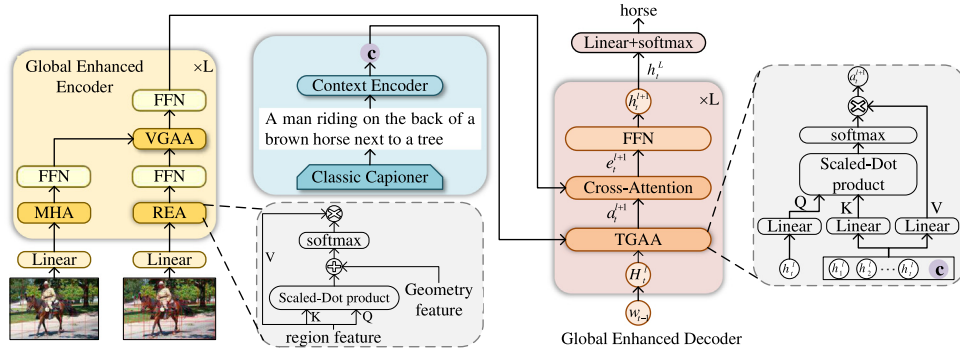


Fig. 3. The overall architecture of our DGET. Note that the residual connect and layer moralization are ignored for simplicity.

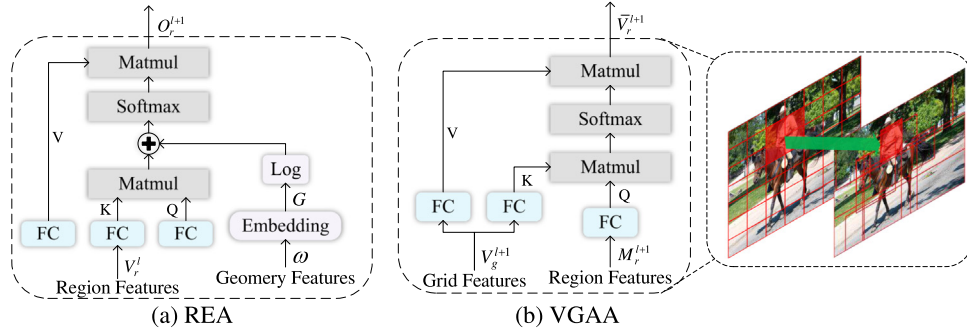


Fig. 4. The relation enhanced attention module and visual global adaptive attention module in our Global Enhanced Encoder.

### 3.2.3. Visual global adaptively attention

We view the grid features as visual global information and adaptively fuse it into region features at each encoder layer. This process is done by our Visual Global Adaptively Attention module. As seen in Fig. 4(b), we take the grid-level representation as key and value vector and regard the region-level representation as the query vector in the multi-head attention module. In this way, each region adaptively captures each fragment of the grid features to realize the complementary of visual information from two different sources feature:

$$\tilde{V}_r^{l+1} = \text{MHA}(M_r^{l+1}, V_g^{l+1}, V_g^{l+1}) \quad (14)$$

### 3.2.4. Global enhanced decoder

In the decoding phase, we denote  $w_t \in \mathbb{R}^d$  as the vector representation of the  $t$ th word, which is the sum of word embedding and positional encoding. Therefore, the input matrix for time step  $t$  is  $W_{t-1} = \{w_0, w_1, \dots, w_{t-1}\} \in \mathbb{R}^{d \times t}$ . We denote the embedded existing caption generated by the classic captioning model as  $P = \{p_1, p_2, \dots, p_k\} \in \mathbb{R}^{d \times t}$ , where  $k$  is the length of existing caption. Similar to the GEE, GED consists of  $L$  identical layers. Before decoding, we encode the existing caption via the proposed Context Encoder to obtain a context vector corresponding to input image.

$$c = \text{ContextEncoder}(P) \quad (15)$$

Where  $c \in \mathbb{R}^{d \times 1}$  is the context vector which can be viewed as sentence embedding of the existing caption. The detail of Context Encoder will be described in Section 3.2.5.

Suppose that the decoder is in the generation process of the  $t$  time step, the input of  $(l+1)$ -th layer is  $H_t^l = \{h_1^l, h_2^l, \dots, h_t^l\} \in \mathbb{R}^{d \times t}$  and context vector  $c$ . They are fed into our proposed Textual Global Adaptive Attention (TGAA) module to model the relationship for the left-hand sequences and adaptively capture global information simultaneously. Subsequently, we denote the output of TGAA as  $a_t^{l+1}$ , and passed it into the multi-head cross-attention

to incorporate with visual feature. Finally, we fed the output of multi-head cross-attention (denote as  $e_t^{l+1}$ ) into a feed-forward neural network (FFN).

$$a_t^{l+1} = \text{Add\&Norm}(\text{TGAA}(h_t^l, H_t^l, c)) \quad (16)$$

$$e_t^{l+1} = \text{Add\&Norm}(\text{MHA}(a_t^{l+1}, V_r^l, V_r^l)) \quad (17)$$

$$h_t^{l+1} = \text{Add\&Norm}(\text{FFN}(e_t^{l+1})) \quad (18)$$

The detail of TGAA will be described in Section 3.2.6. Note that the operation of Add&Norm is the same as that in the encoder.

After  $L$  identical decoder block, the output  $h_t^L$  is fed into the classifier layer over vocabulary to predict the next word. We denote the predicted sentences at previous time step  $t-1$  as  $Y_{t-1} = \{y_0, y_1, \dots, y_{t-1}\}$ , where  $y_i \in V$ , and  $V$  is the vocabulary of the captions. Then the conditional probability distribution of words at current time  $t$  is  $p(y_t|Y_{t-1})$ , which can be calculated by:

$$p(y_t|Y_{t-1}) = \text{softmax}(W_y h_t^L) \quad (19)$$

Where  $W_y \in \mathbb{R}^{|V| \times d}$ , and  $|V|$  is the number of words in the vocabulary.

### 3.2.5. Context encoder

The embedding vectors obtained by different sentence embedding methods have different representations. Hence, we design three alternative structures for Context Encoder to encode the caption, i.e., LSTM-LAST, BiLSTM-MAX and Transformer-Encoder.

**LSTM-LAST.** LSTM is widely used in sequence data encoding, which incorporates suitable information from the current time step via the input gate and forgets useless information from the previous step via the forget gate. We take the last hidden state of LSTM output as a context vector to represent the sentence:

$$c = h_k = \text{LSTM}(p_1, \dots, p_k) \quad (20)$$

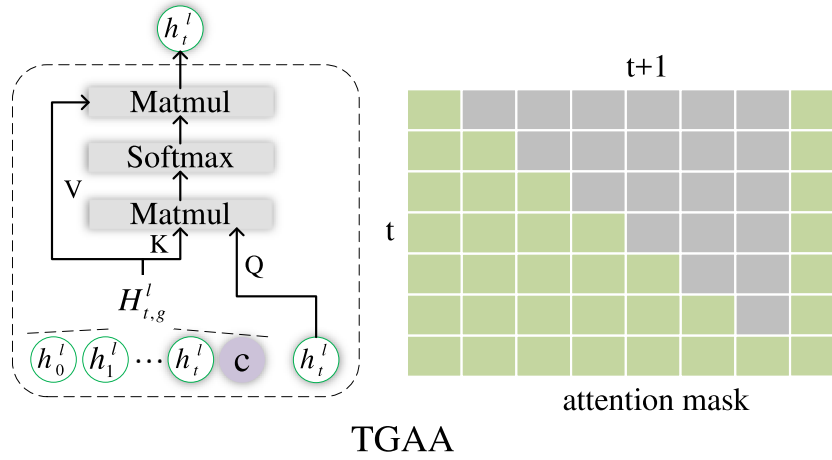


Fig. 5. The calculation flow of the proposed textual global attention module, where  $c$  is context vector.

**BiLSTM-MAX.** A more sophisticated method is to use the Bidirectional LSTM, in which the sentences are encoded forward and backward with LSTM, respectively. We concatenate the two groups of hidden states and select the maximum value over each dimension of the hidden units as context vector:

$$\begin{aligned} \vec{h} &= \text{LSTM}(p_1, \dots, p_k) \\ \overleftarrow{h} &= \text{LSTM}(p_1, \dots, p_k) \\ h &= [\vec{h}; \overleftarrow{h}] \\ c &= \text{MaxPooling}(h) \end{aligned} \quad (21)$$

**Transformer-Encoder.** We also attempt to embed the sentences through the self-attention module. The encoder contains six stacked identical layers, consisting of a multi-head self-attention module and a feed-forward module. Following BERT (Devlin, Chang, Lee, & Toutanova, 2018) and ViT (Dosovitskiy et al., 2020), we add an additional token CLS at the first position of the input sequence and regard the CLS of the last encoding layer as a context vector. We first set  $\text{CLS} = \frac{1}{k} \sum_{i=1}^k p_i$ , then pass  $P_g = (P; \text{CLS})$  into the transformer-encoder structure:

$$\begin{aligned} \bar{P}_g^{l+1} &= \text{Add\&Norm}(\text{MHA}(P_g^l)) \\ P_g^{l+1} &= \text{Add\&Norm}(\text{FFN}(\bar{P}_g^{l+1})) \\ c &= P_{g,0}^L \end{aligned} \quad (22)$$

After multi-layer encoding, we view the token CLS on the output of the last encoder layer as context vector  $c$ .

### 3.2.6. Textual global adaptive attention

Fig. 5 illustrates the calculation flow of our textual global attention module. As shown on the left in Fig. 5, we extend the calculation of self-attention in the decoder by adding the context vector  $c$  to the key and value matrices. In this case, the query vector at each time step captures not only the information of subsequence but also the textual global information by taking a weighted sum of hidden state  $H_t^l$  and context vector  $c$ . We first set  $H_{t,g}^l = (H_t^l; c)$ . The calculation of TGAA as below:

$$a_t^{l+1} = \text{MHA}(h_t^l, H_{t,g}^l, H_{t,g}^l) \quad (23)$$

Note that in order to capture the context vector  $c$  while not exposing future information in training, we insert an extra column in the attention mask, as shown on the right side of Fig. 5.

### 3.3. Training

Following a general practice in image captioning (Anderson et al., 2018; Rennie, Marcheret, Mroueh, Ross, & Goel, 2017),

we first pre-train our model with a token-level Cross-Entropy loss (XE) and then fine-tune the sequence generation using reinforcement learning. Given ground truth sequence  $y_{1:T}^*$  and a captioning model with parameters  $\theta$ . We optimize the following cross-entropy (XE) loss:

$$L_{XE} = - \sum_{t=1}^T \log(p_\theta(y_t^* | y_{1:t-1}^*)) \quad (24)$$

Following previous works (Anderson et al., 2018; Huang et al., 2019; Rennie et al., 2017), we use the CIDEr-D score as reward, as its performance is well. Ref. Mnih and Rezende (2016), we set the number of samples to  $k$ , and we baseline the reward using the mean of the rewards rather than greedy decoding as done in previous methods (Rennie et al., 2017), as we found that it slightly improved the performance. The final gradient expression for one sample is thus:

$$\nabla_\theta L_{RL}(\theta) = - \frac{1}{k} \sum_{i=1}^k ((r(w^i) - b) \nabla_\theta \log p_\theta(w^i)) \quad (25)$$

where  $k$  is the number of samples,  $w^i$  is the  $i$ th sentence sampled from probability distribution,  $r(\cdot)$  is the reward function, and  $b$  is the baseline as below:

$$b = \frac{1}{k-1} \sum_{j=1, j \neq i}^k r(w^j) \quad (26)$$

## 4. Experimental setup

### 4.1. Dataset

The MS COCO is the most widely used dataset in image captioning. The whole MS COCO dataset contains 164,062 images, including 82,783 training images, 40,504 validation images, and 40,775 testing images. Each image is annotated with at least five captions by different Amazon Mechanical Turk (AMT) workers, and the annotation of the training set and validation set is in public. In offline evaluation, we adopt the Karpathy splits (Karpathy & Fei-Fei, 2015) that have been used extensively for reporting results in previous works. This split contains 123,287 images with available annotation, i.e., training set and validation set, including 113,287 images for training, and 5000 images for validation and 5,000 for testing. The online evaluation is done on the MS COCO test server, for which ground truth annotations are not publicly available.

## 4.2. Evaluation metrics

To evaluate the quality of generated caption, lots of automatic evaluation metrics based on the similarity between ground truth and generated captions are proposed. Following the MS COCO captioning challenges 2015, we report results on the four metrics: BLEU (Papineni, Roukos, Ward, & Zhu, 2002), METEOR (Banerjee & Lavie, 2005), ROUGE-L (Lin, 2004), and CIDEr (Vedantam, Lawrence, Zitnick, & Parikh, 2015) in our experiments. Besides, we also report results using the new metric SPICE (Anderson, Fernando, Johnson, & Gould, 2016), which was found to better correlate with human judgments.

## 4.3. Implementation details

**Data preprocessing.** In the experiments, we apply the standard preprocessing practice to the images and captions.

For images, we used two different kinds of visual feature extractors. We utilize the bottom-up features provided by Anderson et al. (2018) as region-level features. Each image contains 10–100 regions, where each region is represented as a 2048-dimensional vector. At the same time, we use employ pre-trained ResNet-152 (He et al., 2016) to extract grid-level image features. All images are resized to  $224 \times 224$  before being fed into the CNN. The grid features are the outputs of the last convolutional layer of ResNet-152, with the size of  $7 \times 7 \times 2048$ .

For captions, we convert each sentence to lower case and discard all the non-alphabetic characters. We remove the words that occur less than 5 times in the training set, and replace it with token  $\langle unk \rangle$ . We insert an  $\langle start \rangle$  and  $\langle end \rangle$  at the beginning and end of each sentence to determine where the sentence begins and ends. In addition, we set the maximum sentence length to 16. If the sentence exceeds length limitation, it will be truncated, otherwise it will be filled with special token  $\langle pad \rangle$ . Finally, the vocabulary size is 9487. Each word is then embedded into a  $d$ -dimensional word embedding space.

For existing caption, we use our reproduced Up-Down captioner (Anderson et al., 2018) and publicly available pre-trained AoA model as our classic captioner, which performance results can be seen in Table 3.

**Training details.** In our implementation, we set  $d$  to 512 and the number of head to 8. We employ dropout with a keep probability of 0.9 after each attention and feed-forward layer. The number of layers for encoder and decoder is set to 3 and 6. In the cross-entropy(XE) pre-training stage, we following the learning rate scheduling strategy with a warmup equal to 20,000 iterations. After the 20 epoch XE pre-training stage, we start to optimize our model with CIDEr reward with  $5 \times 10^{-6}$  learning rate and 30 epoch. The number of samples  $k$  is set 5. We train all models using the Adam optimizer (Kingma & Ba, 2014) with betas=(0.9, 0.999), and the batch size is set 50. Note that we do not fine-tune the visual feature extractor and classic captioner during training. All the experiments are implemented in PyTorch 1.4 with CUDA version 9.2. The training time for each two-stage model to run on NVIDIA Tesla P100 GPU was about 1 + 4 days.

## 5. Experimental results and analysis

### 5.1. Comparison with state-of-the-art methods

**Offline Evaluation.** Table 1 shows the performance of the state-of-the-art models and our approach on the offline COCO Karpathy and Fei-Fei (2015) test split. The compared models include: SCST (Rennie et al., 2017), Up-Down (Anderson et al., 2018), GCN-LSTM (Yao et al., 2018), SGAE (Yang et al., 2019), SRT (Wang et al., 2020), AoA (Huang et al., 2019), ORT (Herdade

et al., 2019), ETA (Li, Zhu, et al., 2019), M2 (Cornia et al., 2020), NG-SAN (Guo et al., 2020) and GET (Ji et al., 2021). We present the results of the proposed DGET with three different Context Encoder, denoted by Ours-LSTM-LAST, Ours-BiLSTM-MAX, and Ours-Transformer-Encoder.

LSTM-based methods (Anderson et al., 2018; Huang et al., 2019; Wang et al., 2020; Yang et al., 2019; Yao et al., 2018) usually contain only two layers of LSTM, namely attention lstm and language lstm, and the simple structure limits their performance to some extent. On the contrary, our Transformer-based model is capable of capturing the intra-modal relationship by a large amount of self-attention modules to achieve superior performance in the quality of generated sentences.

Most similar to our work, GET (Ji et al., 2021) learns an additional global vector at each layer of the encoder by self-attention and uses an lstm to refine the global vector from multiple layers to obtain the final visual global representation. The visual global representation is concatenated with the output of the encoder to allow the decoder to capture both local and global visual information at each time step. In contrast to GET, our model focuses on using visual global information to enhance the representation of local information (i.e., utilizing grid-level features to enhance region-level features). Specifically, we acquire grid and region features through two different visual feature extractors (i.e., ResNet and Faster RCNN), and exploit the complementarity of them to obtain a stronger visual representation, which is essentially different from GET in that its global representation is an additional vector obtained by learning. Moreover, we explored the textual global information that is ignored by GET.

For other Transformer-based methods, ORT (Herdade et al., 2019) explicitly incorporates information about the spatial relationship between input detected regions in Transformer architecture. ETA (Li, Zhu, et al., 2019) extend Transformer model to exploit complementary information of visual regions and semantic attributes simultaneously by introducing the EnTangled Attention and Gated Bilateral Controller. NG-SAN (Guo et al., 2020) model improves the scene understanding of the model by reducing the covariate shift problem and explicitly using geometric information in the self-attention module. M2 (Cornia et al., 2020) exploits prior knowledge through learning a set of memory vectors and proposes a meshed connectivity between encoding and decoding modules. Our proposed DGET focuses on capturing visual and textual global information simultaneously at the encoder and decoder to boost the performance of image captioning.

The result indicates that our method surpasses all Transformer-based approaches in terms of BLEU-4, ROUGE-L, CIDEr and SPICE, and achieves competitive performance on BLEU-1 and METEOR compared to the SOTA approach. In particular, it advances the current state-of-the-art on CIDEr and BLEU-4 by 0.3 and 0.4 points, respectively.

**Online Evaluation.** To make a complete comparison with other published methods, we have submitted DGET with three different context encoders to MS COCO online evaluation server. Table 2 reports the performance of our proposed model and other top-ranking published models on the MS COCO test server. Note that the ensemble model always has better performance. For fair comparison, we use the single model to compare with the published state-of-the-art models (Anderson et al., 2018; Guo et al., 2020; Li, Zhu, et al., 2019; Yang et al., 2019; Yao et al., 2018). Two evaluation settings are provided: c5 means five caption references per image, and c40 means forty caption references per image. As can be seen from the table, compared with the published models, our single model significantly outperforms all the other models in terms of BLEU-1(c40), BLEU-2(c5, c40), BLEU-3(c5, c40), BLEU-4(c5, c40), ROUGE(c5, c40) and CIDEr(c5, c40), and achieves competitive performance on BLEU-1(c5) and METEOR(c5, c40).

**Table 1**

Performance comparisons on MS-COCO Karpathy test split under the REINFORCE optimization stage. \* indicates the results obtained from the publicly available pre-trained model. † means results from our reproduction. All values are reported as percentage (%).

Model	B-1	B-4	M	R	C	S
SCST (Rennie et al., 2017)	–	34.2	26.7	55.7	114.0	–
Up-Down (Anderson et al., 2018)	79.8	36.3	27.7	56.9	120.1	21.4
Up-Down†(Anderson et al., 2018)	79.4	36.7	27.9	57.6	122.7	21.4
GCN-LSTM (Yao et al., 2018)	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
SRT (Wang et al., 2020)	80.3	38.5	28.7	58.4	129.1	22.4
AoA (Huang et al., 2019)	80.2	38.9	29.2	58.8	129.8	22.4
AoA*(Huang et al., 2019)	80.5	39.1	29.0	58.9	128.9	22.7
ORT (Herdade et al., 2019)	80.5	38.6	28.7	58.4	128.3	22.6
ETA (Li, Zhu, et al., 2019)	<b>81.5</b>	39.3	28.8	58.9	126.6	22.7
M2(Cornia et al., 2020)	80.8	39.1	29.2	58.6	131.2	22.6
NG-SAN (Guo et al., 2020)	–	39.9	<b>29.3</b>	59.2	132.1	<b>23.3</b>
GET (Ji et al., 2021)	<b>81.5</b>	39.5	<b>29.3</b>	58.9	131.6	22.8
Ours-LSTM- LAST	81.2	39.9	29.2	59.2	132.0	23.1
Ours-BiLSTM- MAX	81.3	40.1	29.2	59.2	132.1	23.1
Ours-Transformer-Encoder	81.3	<b>40.3</b>	29.2	<b>59.4</b>	<b>132.4</b>	<b>23.3</b>

**Table 2**

Performance comparisons with other advanced single models on the official MS COCO test server. All values are reported as percentage (%).

Model	B-1		B-2		B-3		B-4		M		R		C	
	c5	c40	c5	c40	c5	c40	c5	c40	c5	c40	c5	c40	c5	c40
Up-Down (Anderson et al., 2018)	80.2	<b>95.2</b>	64.1	88.8	49.1	79.4	36.9	68.5	27.6	36.7	57.1	72.4	117.9	120.5
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)	80.6	95.0	65.0	88.9	50.1	79.6	37.8	68.7	28.1	37.0	58.2	73.1	122.7	125.5
ETA (Li, Zhu, et al., 2019)	<b>81.2</b>	95.0	65.5	89.0	50.9	80.4	38.9	70.2	28.6	38.0	58.6	73.9	122.1	124.4
NG-SAN (Guo et al., 2020)	80.8	95.0	65.4	89.3	50.8	80.6	38.8	70.2	<b>29.0</b>	<b>38.4</b>	58.7	74.0	126.3	128.6
Ours-LSTM-LAST	80.9	95.0	<b>65.7</b>	89.5	<b>51.2</b>	80.8	<b>39.2</b>	70.4	28.8	38.1	58.8	74.1	<b>126.6</b>	128.5
Ours-BiLSTM-MAX	80.9	<b>95.2</b>	<b>65.7</b>	89.5	<b>51.2</b>	80.8	<b>39.2</b>	70.4	28.9	38.2	58.8	74.0	<b>126.6</b>	128.2
Ours-Transformer-Encoder	80.8	95.1	65.6	<b>89.6</b>	51.1	<b>81.3</b>	39.1	<b>71.2</b>	28.9	<b>38.4</b>	<b>58.9</b>	<b>74.4</b>	126.3	<b>129.2</b>

**Table 3**

Ablation on different variants of our DGET. – represent the original Transformer block. All results are reported after the REINFORCE optimization stage.

Encoder	Decoder	B-1	B-4	M	R	C	S
–	GED	80.8	39.3	28.9	58.9	130.2	22.4
w/REA	GED	80.9	39.7	29.0	59.0	130.5	22.6
w/VGAA	GED	81.1	39.9	29.1	59.0	131.3	22.8
GEE	–	81.1	39.8	29.0	59.0	131.7	22.8
GEE	GED	81.2	39.9	29.2	59.2	132.0	23.1

## 5.2. Ablation studies

To validate the effectiveness of our proposed DGET, we conduct extensive ablation experiments to study the contributions of each module.

### 5.2.1. The effect of GEE

To better demonstrate the effectiveness of GEE, we conduct several ablative experiments. For a fair comparison, we use the same GED (the Context Encoder is LSTM-LAST, and the existing captions are generated by our replication of Up-Down captioning model). We choose the plain Transformer with GED as the baseline, which is shown in the 1st line in Table 3. Then, we investigate the impact of the VGAA and REA on the captioning performance for the GEE, the results as shown in lines 2, 3 of Table 3. Benefit from explicitly leveraging the objects' geometric information, REA can slightly improve the performance from 130.2 to 130.5 on CIDEr score. We can observe that using VGAA to adaptively fuse grid features into region features results in a significant improvement compared to the baseline on CIDEr by 1.1 points. When using our proposed GEE (i.e. REA+VGAA), we obtain the desired performance with 1.8 points CIDEr score improvement. This further demonstrates the effectiveness of using grid-level features to enhance region-level features and explicitly using geometric information.

### 5.2.2. The effect of GED

We conducted several experiments to demonstrate the effectiveness of each module in our GED. To make a fair comparison, we fixed the encoder structure (i.e., VGAA+REA). First, we explore the performance impact of three different structures of Context Encoder, the last three rows of Table 1 show the results of the ablation experiment. We can observe that LSTM-LAST is slightly worse than BiLSTM-MAX since LSTM-LAST cannot capture long-distance dependencies, which fails to generate global representation with rich semantic information. In contrast, BiLSTM-MAX encodes the caption from forward and backward, respectively, which solves the problem to a certain extent and gain better performance. On the other hand, the Transformer Encoder method has better performance in all metrics compared to BiLSTM-MAX. It is demonstrated that the self-attention module can capture more critical information than LSTM.

Second, to study the contribution of context vector to predict word, we visualized the attention weight of TGAA module to context vector in each layer of decoder at each time step. As shown in Fig. 6, the focusing intensity of context vector decreases significantly with the generation process, which demonstrates that context vector is committed to providing textual global information for the decoder in the early time step. When the generated sentence is gradually complete, the decoder has enough semantic information to generate accurate words. At this time, the model does not need to attend the context vector. In addition, the attention values to context vector in the shallow layer (i.e., layer1 and layer2) are always higher than other layers. This phenomenon may be because the model focuses on learning the knowledge brought by the existing captions in the shallow layers and then learns more semantic representations based on it in the deep layer. Finally, by comparing the fourth and Fifth rows of Table 3, we observe that GED brings a 0.3 points improvement on CIDEr score, which further demonstrates the effectiveness of explicitly leveraging textual global information.

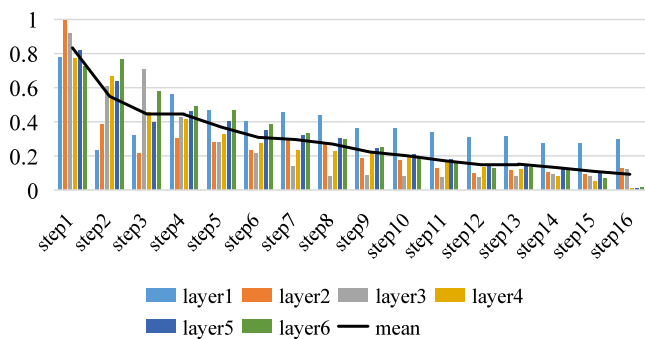


**Table 4**  
Ablation on different of existing caption after the Cross-Entropy training.

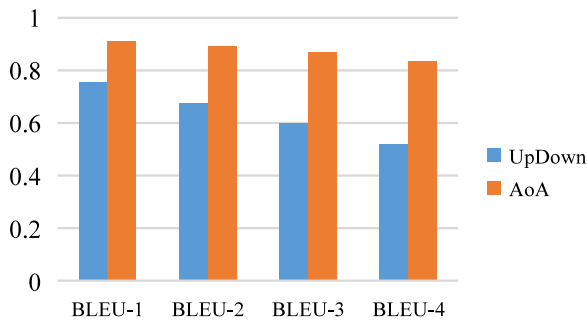
Context Encoder	Caption	B-1	B-4	M	R	C	S
LSTM-LAST	Up-Down	76.3	36.4	28.1	56.8	116.0	21.1
	AoA	76.9	36.2	28.1	56.9	117.0	21.5
BiLSTM-MAX	Up-Down	77.0	35.7	27.7	56.6	115.7	21.1
	AoA	77.2	37.0	28.1	57.2	117.0	21.2
Transformer Encoder	Up-Down	76.5	36.2	27.8	56.5	115.5	20.9
	AoA	76.9	36.8	28.2	57.0	116.9	21.3

**Table 5**  
Ablation on different of existing caption after the REINFORCE optimization stage.

Context Encoder	Caption	B-1	B-4	M	R	C	S
LSTM-LAST	Up-Down	81.2	39.9	29.2	59.2	132.0	23.1
	AoA	81.1	39.7	29.1	59.0	130.6	22.9
BiLSTM-MAX	Up-Down	81.3	40.1	29.2	59.2	132.1	23.1
	AoA	81.1	39.8	29.2	59.0	130.9	22.9
Transformer Encoder	Up-Down	81.3	40.3	29.2	59.4	132.4	23.3
	AoA	81.2	39.8	29.2	59.2	131.1	23.0



**Fig. 6.** The visualization of attention weight of context vector at each time step. layer-1 to layer-6 denote the TGAA modules of layers 1 to 6 in the decoder, respectively. All reported values are from the offline Karpathy testing set.



**Fig. 7.** The sentences similarity between generated by our model and the original captioner.

### 5.2.3. The effect of existing caption

In order to explore the impact of the different existing captions on performance, we use the existing captions generated by Up-Down model and AoA model, respectively, to conduct several experiments. In addition, we also consider the sensitivity of three different variants of Context Encoder to different existing captions. Table 4 shows the results after XE training, we can observe that the performance of the model using AoA captions is slightly better than using UpDown captions on all evaluation metrics. The reason is that AoA captions contain more accurate content and rich semantic information, which helps model fitting. However, the result after CIDEr optimization is opposite. As shown in Table 5, the models using AoA captions as the existing caption are not as good as those using Up-down captions. In order to

explain this phenomenon, we visualized the similarity between the caption generated by our model and that generated by the original model. Concretely, we use BLEU (Papineni et al., 2002) to represent the similarity between two sentences and use the BiLSTM-MAX as Context Encoder, the results are shown in Fig. 7. We can be seen that the sentences generated by the AoA-base model are highly similar to AoA captions, with values above 80 for the BLEU (1–4) metrics. This result shows that our model is imitating AoA captions rather than learning how to use them. The reason may be that high-quality AoA captions (128.9 CIDEr score in the publicly available pre-trained model) guide the optimizer to update the model's parameters in a direction close to it during CIDEr optimization. To sum up, the UpDown captions is suitable as an existing caption, because it contains the general semantics of the image, but it is not enough to have a significant impact during training.

### 5.3. Qualitative analysis

#### 5.3.1. Qualitative examples




In Fig. 8, we show several qualitative examples of the captions generated by our DGET on MS COCO dataset. To better analyze the practical performance of our model, we focus on exhibiting the effect of different qualities of the existing captions on the final generated results. Firstly, in the first and second examples of the figure, we can see that the existing captions are somewhat relevant to image content and logically correct. As in the first example, “a woman is eating a piece of cake on a plate”, which is highly consistent with the content of the image. However, it is still missing some detailed information, i.e. “a woman in a wedding dress”, which is captured by our DGET. In the second example, the sentence “elephants standing next to a man” is almost semantically correct for the image, which can be regarded as another perspective of understanding the figure. However, this image may tell us more than that. In contrast, our DGET is able to generate more accurate and descriptive captions with more fine-grained information for the image, i.e. “people watching elephants in a zoo”. Secondly, as can be seen in the third and fourth lines, our model is capable of replacing the incorrect state and attribute word when most of the existing captions are correct. For example, changing ‘yellow’ to ‘orange’ and “brushing” to “blow drying”. Finally, in the last line, we can observe that our model still generates accurate descriptions (101.6 CIDEr score) even when the existing caption is incorrect (0.5 CIDEr score).

Furthermore, we performed an additional comparison of the GET (Ji et al., 2021), which is similar to our work and has obtained advanced performance. The result is shown in Fig. 9. Although GET brings high quality descriptions by explicitly learning an additional visual representation, it still lacks descriptions of parts of the scene, e.g. “airport runway” in Example 1. In Example 2, GET gets detailed information about the object's attributes (i.e., blue and white stripes), but lacks information about the object's state (i.e., sitting on top of a field) compared to our method. In Example 3, the GET-generated sentence is very similar to ours, but it has the fatal error of misidentifying the number of objects. The reason might be the bias of the datasets, since about 76% of the sentences in MSCOCO dataset start with ‘a’ or ‘an’. In contrast, our approach successfully overcomes this difficulty by adaptively capturing textual global information.

To sum up, we can observe three major points. First, when the existing captions contain most of the semantic content of the image (i.e., high-quality existing caption), our DGET can refine it to generate a fine-grained caption. Second, when the existing captions are partially correct (i.e., moderate-quality existing caption), our model focuses on modifying the incorrect words through the guidance of visual information to obtain the semantically

	<p>GT1: there are two people enjoying a wedding reception.</p> <p>GT2: a woman in a wedding dress with another woman in a suit behind.</p> <p>GT3: a woman in a wedding dress with another lady holding a piece of cake.</p> <p>GT4: a red head girl holding a piece of cake.</p> <p>GT5: a bride is with a long red haired person with cake.</p> <p>Caption: a woman is eating a piece of cake on a plate. # CIDEr 102.7</p> <p>Ours: a woman in a <b>wedding dress</b> eating a piece of cake. # CIDEr 245.1</p>
	<p>GT1: several elephants are in a habitat as heads are in the foreground.</p> <p>GT2: a small gray elephant standing in an exhibit at a zoo.</p> <p>GT3: people are watching four elephants in a zoo.</p> <p>GT4: several elephants in zoo enclosure with onlookers watching.</p> <p>GT5: an elephant in a zoo stands in front of the crowd.</p> <p>Caption: a <b>group of elephants standing next to a man</b>. # CIDEr 52.6</p> <p>Ours: a <b>group of people watching elephants in a zoo</b>. # CIDEr 166.4</p>
	<p>GT1: a woman blow drying her hair, in he mirror.</p> <p>GT2: a woman stands in a bathroom blow drying her hair</p> <p>GT3: a woman blow drying her hair in a room with a window.</p> <p>GT4: a young woman blow drying her hair in a bathroom.</p> <p>GT5: a girl in a bathroom blow-drying her hair.</p> <p>Caption: a woman is <b>brushing</b> her hair with a hair dryer. # CIDEr 87.1</p> <p>Ours: a woman <b>blow drying</b> her hair <b>in a bathroom</b>. # CIDEr 604.7</p>
	<p>GT1: a woman is riding an orange sports motorcycle.</p> <p>GT2: large woman on the back of on orange motorcycle.</p> <p>GT3: a woman riding an orange motorcycle on a race track.</p> <p>GT4: a large woman is riding a motor bike.</p> <p>GT5: a smiling woman on a bright orange motorcycle.</p> <p>Caption: a woman sitting on a <b>yellow</b> motorcycle. # CIDEr 92.0</p> <p>Ours: a woman sitting on an <b>orange</b> motorcycle. # CIDEr 212.7</p>
	<p>GT1: a cheeseburger is pictured on a tray next to a cup.</p> <p>GT2: a yummy looking hamburger with cheese, tomatoes, and lettuce.</p> <p>GT3: a cheese burger that has some lettuce and tomato on it.</p> <p>GT4: a large double cheeseburger with lettuce and tomato.</p> <p>GT5: a hamburger covered in cheese and veggies and a bun.</p> <p>Caption: a sandwich <b>on a plate with a table</b>. # CIDEr 0.5</p> <p>Ours: a sandwich with <b>lettuce and cheese</b> on a plate. # CIDEr 101.6</p>

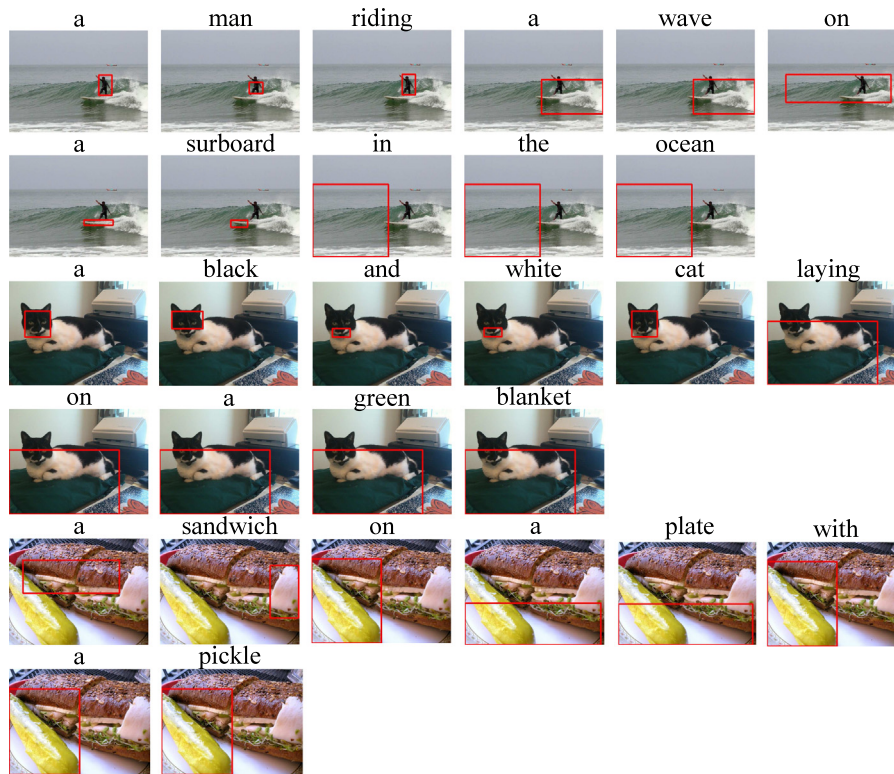
**Fig. 8.** Examples of the image captioning results by our proposed DGET and classic captioner with ground truth sentences and the corresponding CIDEr scores. Some detailed and accurate words are marked in green, the wrong words are marked in red, and the inaccurate words are marked in yellow. The comparison between DGET and existing captions shows that our model captures crucial semantic information from coarse caption, and refine it to form more comprehensive content captions.

	<p>GT1: a large jetliner sitting on top of an airport runway.</p> <p>GT2: airline employees by an aircraft parked at the gate.</p> <p>GT3: the plane is parked at the gate at the airport terminal.</p> <p>GT4: a large jetliner sitting on top of an airport runway.</p> <p>GT5: a large white airplane and a person on a lot.</p> <p>GET: a large airplane parked at the airport with a man. # Blue-4: 0 CIDEr: 121.7</p> <p>Ours: a large airplane parked on top of an <b>airport runway</b>. # Blue-4: 61.4 CIDEr: 157.6</p>
	<p>GT1: a small blue plane sitting on top of a field.</p> <p>GT2: an E2 airplane painted blue with black and white stripes.</p> <p>GT3: an old warplane is on display in a field.</p> <p>GT4: a blue small plane standing at the airstrip.</p> <p>GT5: model airplane with an American insignia and stripes on wings.</p> <p>GET: a small plane is painted with blue and white stripes. # Blue-4: 0 CIDEr: 123.4</p> <p>Ours: a small plane <b>sitting on top of a field</b>. # Blue-4: 83.0 CIDEr: 182.3</p>
	<p>GT1: a few drag queens make some cake and eat it.</p> <p>GT2: two people wearing wigs holding a cake that has their picture on it.</p> <p>GT3: cake with a picture of two girls holding it in the picture.</p> <p>GT4: two women wearing wigs holding a cake with their picture on it.</p> <p>GT5: a couple of people are holding up a cake.</p> <p>GET: a woman is holding a cake with a picture on it. # Blue-4: 30.0 CIDEr: 188.0</p> <p>Ours: <b>two</b> woman holding a cake with a picture on it. # Blue-4: 59.6 CIDEr: 197.1</p>

**Fig. 9.** Examples of captions generated by our proposed DGET and GET (Ji et al., 2021) as well as the corresponding ground truth sentences, Blue-4 and CIDEr scores. Some detailed and accurate words are marked in green, the wrong words are marked in red..

correct caption. Finally, when the existing captions are mostly wrong (i.e., low-quality existing captions), the powerful visual representation enables our model to generate correct sentences

and eliminate its influence. These examples demonstrate that our approach is not simply to imitate an existing description, but to learn a better expression based on it. Moreover, the advantage of



**Fig. 10.** Visualization of visual attention weights for generated captions.

**Table 6**

Results of human evaluation on 500 images randomly sampled from MSCOCO Karpathy test split.

Model	better	worse	indistinguishable
Ours vs Up-Down	24%	14%	62%
Ours vs AoANet	18%	12%	70%

capturing visual and textual global information simultaneously in the transformer structure is further demonstrated by comparison with GET model.

### 5.3.2. Human evaluation

As the automatic evaluation metrics (e.g. BLEU and CIDEr) do not necessarily consistent with human judgment, we additionally conduct a human evaluation to evaluate our method against two baselines, i.e. Up-Down and AoANet. We randomly sampled 500 images from the Karpathy test split and invited 20 different workers who have prior experience with image captioning for human evaluation. We showed them each image and the corresponding captions generated by the baseline and asked them to make the comparison with the sentences generated by our model based on relevance and coherence. The results of the comparisons on two different baselines are shown in Table 6. We can see from the table that, our DGET significantly outperforms the baselines.

### 5.4. Attention analysis

In order to better understand the effectiveness of our model, we investigate the contribution of attention weights learned by attention module in GEE and GED to the model output.

**Attention of the Encoder.** In order to explore the influence of each visual region on the generated words, we average attention weights of 8 heads of the Cross-Attention module in the last layer of decoder at each time step. As shown in Fig. 10, our

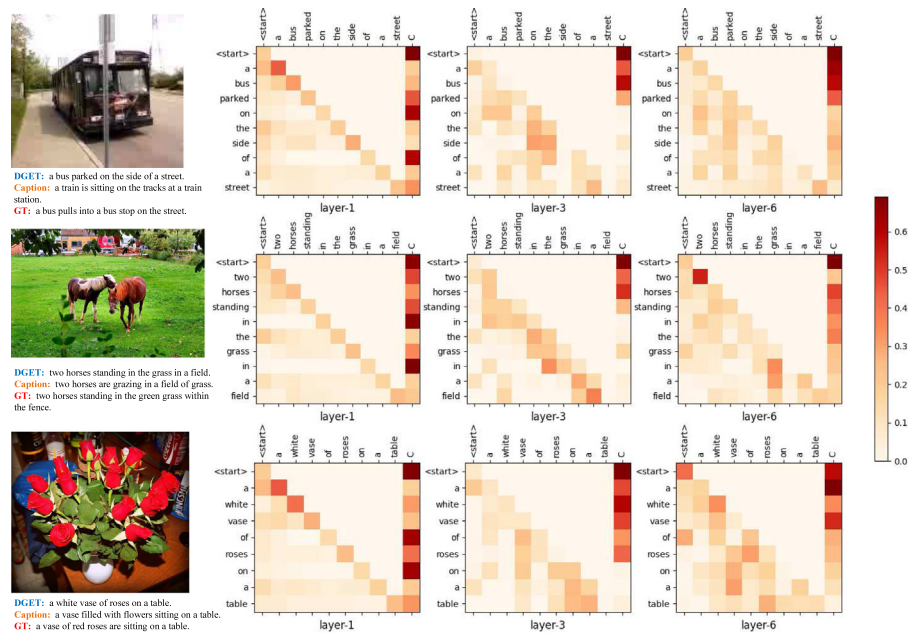
approach correctly grounds image regions to words at each time step. As in example 1, the area representing the “a man riding... on a surfboard” in the figure is tiny, our model can recognize it correctly, which further demonstrates that our approach is capable of capturing fine-grained details and relationships between objects.

**Attention of the Decoder.** Our GED uses six identical attention blocks. For simplicity, we visualize the attention weight matrix of the TGAA module for the first attention block, third attention block and the last attention block, which represent the shallow, middle and deep layers of the decoder, respectively. We also set low quality (i.e., the first example) and high quality (i.e., the second and third examples). As shown in Fig. 11, we find that the largest attention value in Layer-1 almost appears on the context vector, which indicates that our GED tends to capture the semantic content of the context vector at a shallow level. In contrast, the attention weights in layer-3 and layer-6 do not focus heavily on the context vector, indicating that our GED captures other semantic information at a deeper level. Besides, the attention weight to the context vector decreases significantly with the increase of time step, which indicates that the model gradually does not need the guidance of the textual global information when the semantic content of the generated sentences tends to be complete. By comparing Example 1 with Examples 2 and 3, we can observe that the attention values for context vectors encoded by low-quality are significantly lower than for high-quality ones, which shows that our model does not blindly follow the Context vector. These further demonstrate the advantage of our model.

## 6. Conclusions

In this paper, we propose Dual Global Enhanced Transformer (DGET) for image captioning. DGET consists of Global Enhanced Encoder, which leveraging grid-level feature to provide visual





**Fig. 11.** Visualizations of the 1st, 3rd and 6th attention maps (i.e.,  $\text{softmax}(\frac{QK^T}{\sqrt{d_h}})$ ) of TGAA module in the GED. Caption denotes the existing caption generated by classic captioner. GT denotes one of the five ground-truth captions provided by MS COCO. In the tick label, the <start> indicates the beginning signal of generation, and C denotes the context vector.

global information for region-level feature to generate more comprehensive visual representation, and the Global Enhanced Decoder which adaptively fuses textual global information into the decoder to obtain more semantic representation at each time step. We also proposed three different Context Encoders to better explore the existing caption. Extensive results demonstrate the superiority of our approach that achieves a new state-of-the-art on offline test splits.

In the future, we will consider using non-autoregressive methods to avoid the computational costs caused by the classic captioner. In addition, we will consider more different sentence embedding methods as our context encoder to obtain a better context vector, as well as try to combine patch-level features to enhance the visual representation.

## 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 National Natural Science Foundation of China (Nos. 61966004, 61866004), Guangxi Natural Science Foundation, China (No. 2019GXNSFDA245018), Guangxi "Bagui Scholar" Teams for Innovation and Research Project, Guangxi Talent Highland Project of Big Data Intelligence and Application, China, and Guangxi Collaborative Innovation Center of Multi-source Information Integration and Intelligent Processing, China.

## References

- Anderson, P., Fernando, B., Johnson, M., & Gould, S. (2016). SPICE: Semantic propositional image caption evaluation. In *Proceedings of the European conference on computer vision* (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 the IEEE conference on computer vision and pattern recognition* (pp. 6077–6086).
- Ba, J. L., Kiros, J. R., & Hinton, G. E. (2016). Layer normalization. arXiv preprint arXiv:1607.06450.
- 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).
- Chen, Y.-C., Li, L., Yu, L., El Kholy, A., Ahmed, F., Gan, Z., et al. (2020). UNITER: Universal image-text representation learning. In *Proceedings of the European conference on computer vision* (pp. 104–120).
- Chen, L., Zhang, H., Xiao, J., Nie, L., Shao, J., Liu, W., et al. (2017). SCA-CNN: Spatial and channel-wise attention in convolutional networks for image captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 5659–5667).
- Cornia, M., Stefanini, M., Baraldi, L., & Cucchiara, R. (2020). Meshed-memory transformer for image captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 10578–10587).
- Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.
- Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., et al. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929.
- Fang, H., Gupta, S., Iandola, F., Srivastava, R. K., Deng, L., Dollár, P., et al. (2015). From captions to visual concepts and back. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1473–1482).
- Guo, L., Liu, J., Zhu, X., Yao, P., Lu, S., & Lu, H. (2020). Normalized and geometry-aware self-attention network for image captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 10327–10336).
- He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770–778).
- Herdade, S., Kappeler, A., Boakye, K., & Soares, J. (2019). Image captioning: Transforming objects into words. In *Advances in neural information processing systems* (pp. 11137–11147).
- Huang, F., Li, Z., Wei, H., Zhang, C., & Ma, H. (2020). Boost image captioning with knowledge reasoning. *Machine Learning*, 109(12), 2313–2332.
- 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).
- Hudson, D. A., & Manning, C. D. (2019). GQA: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 6700–6709).



- Ji, J., Luo, Y., Sun, X., Chen, F., Luo, G., Wu, Y., et al. (2021). Improving image captioning by leveraging intra-and inter-layer global representation in transformer network. In *Proceedings of the AAAI conference on artificial intelligence* (pp. 1655–1663).
- 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).
- Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.
- 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.
- Li, G., Duan, N., Fang, Y., Gong, M., & Jiang, D. (2020). Unicoder-VL: A universal encoder for vision and language by cross-modal pre-training. In *Proceedings of the AAAI conference on artificial intelligence* (pp. 11336–11344).
- Li, L. H., Yatskar, M., Yin, D., Hsieh, C.-J., & Chang, K.-W. (2019). VisualBERT: A simple and performant baseline for vision and language. arXiv preprint arXiv:1908.03557.
- Li, X., Yin, X., Li, C., Zhang, P., Hu, X., Zhang, L., et al. (2020). OSCAR: Object-semantic aligned pre-training for vision-language tasks. In *Proceedings of the European conference on computer vision* (pp. 121–137).
- 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 *Proceedings of the ACL workshop on text summarization branches out* (pp. 74–81).
- Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., et al. (2014). Microsoft COCO: Common objects in context. In *Proceedings of the European conference on computer vision* (pp. 740–755).
- Liu, F., Liu, Y., Ren, X., He, X., & Sun, X. (2019). Aligning visual regions and textual concepts for semantic-grounded image representations. arXiv preprint arXiv:1905.06139.
- Liu, F., Wu, X., Ge, S., Zhang, X., Fan, W., & Zou, Y. (2020). Bridging the gap between vision and language domains for improved image captioning. In *Proceedings of the ACM international conference on multimedia* (pp. 4153–4161).
- Lu, J., Batra, D., Parikh, D., & Lee, S. (2019). ViLBERT: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. arXiv preprint arXiv:1908.02265.
- Lu, J., Xiong, C., Parikh, D., & Socher, R. (2017). Knowing when to look: Adaptive attention via a visual sentinel for image captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 375–383).
- Mnih, A., & Rezende, D. (2016). Variational inference for monte carlo objectives. In *Proceedings of the international conference on machine learning* (pp. 2188–2196).
- Ordonez, V., Kulkarni, G., & Berg, T. (2011). Im2text: Describing images using 1 million captioned photographs. In *Advances in neural information processing systems* (pp. 1143–1151).
- Pan, Y., Yao, T., Li, Y., & Mei, T. (2020). X-linear attention networks for image captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 10971–10980).
- 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 of the association for computational linguistics* (pp. 311–318).
- Qin, Y., Du, J., Zhang, Y., & Lu, H. (2019). Look back and predict forward in image captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 8367–8375).
- Ren, S., He, K., Girshick, R., & Sun, J. (2016). Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6), 1137–1149.
- 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).
- Sammani, F., & Melas-Kyriazi, L. (2020). Show, edit and tell: A framework for editing image captions. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4808–4816).
- Sharma, P., Ding, N., Goodman, S., & Soricut, R. (2018). Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the annual meeting of the association for computational linguistics* (pp. 2556–2565).
- Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1), 1929–1958.
- Tan, H., & Bansal, M. (2019). LXMERT: Learning cross-modality encoder representations from transformers. arXiv preprint arXiv:1908.07490.
- 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).
- Wang, L., Bai, Z., Zhang, Y., & Lu, H. (2020). Show, recall, and tell: Image captioning with recall mechanism. In *Proceedings of the AAAI conference on artificial intelligence* (pp. 12176–12183).
- Wei, H., Li, Z., Huang, F., Zhang, C., Ma, H., & Shi, Z. (2021). Integrating scene semantic knowledge into image captioning. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 17(2), Article 52.
- Wei, H., Li, Z., Zhang, C., & Ma, H. (2020). The synergy of double attention: Combine sentence-level and word-level attention for image captioning. *Computer Vision and Image Understanding*, 201, Article 103068.
- Wu, Q., Shen, C., Liu, L., Dick, A., & Van Den Hengel, A. (2016). What value do explicit high level concepts have in vision to language problems? In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 203–212).
- Xiong, R., Yang, Y., He, D., Zheng, K., Zheng, S., Xing, C., et al. (2020). On layer normalization in the transformer architecture. In *Proceedings of the international conference on machine learning* (pp. 10524–10533).
- Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhutdinov, R., et al. (2015). Show, attend and tell: Neural image caption generation with visual attention. In *Proceedings of the international conference on machine learning* (pp. 2048–2057).
- 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* (pp. 684–699).
- Yao, T., Pan, Y., Li, Y., & Mei, T. (2019). Hierarchy parsing for image captioning. In *Proceedings of the IEEE international conference on computer vision* (pp. 2621–2629).
- Yao, T., Pan, Y., Li, Y., Qiu, Z., & Mei, T. (2017). Boosting image captioning with attributes. In *Proceedings of the IEEE international conference on computer vision* (pp. 4894–4902).
- You, Q., Jin, H., Wang, Z., Fang, C., & Luo, J. (2016). Image captioning with semantic attention. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4651–4659).
- 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, J., Li, J., Yu, Z., & Huang, Q. (2019). Multimodal transformer with multi-view visual representation for image captioning. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(12), 4467–4480.